

Distributed Recursive Least-Squares: Stability and Performance Analysis[†]

*Gonzalo Mateos, Student Member, IEEE, and Georgios B. Giannakis, Fellow, IEEE**

Submitted: September 23, 2011

Abstract

The recursive least-squares (RLS) algorithm has well-documented merits for reducing complexity and storage requirements, when it comes to online estimation of stationary signals as well as for tracking slowly-varying nonstationary processes. In this paper, a *distributed* recursive least-squares (D-RLS) algorithm is developed for cooperative estimation using ad hoc wireless sensor networks. Distributed iterations are obtained by minimizing a separable reformulation of the exponentially-weighted least-squares cost, using the alternating-minimization algorithm. Sensors carry out reduced-complexity tasks locally, and exchange messages with one-hop neighbors to consent on the network-wide estimates adaptively. A steady-state mean-square error (MSE) performance analysis of D-RLS is conducted, by studying a stochastically-driven ‘averaged’ system that approximates the D-RLS dynamics asymptotically in time. For sensor observations that are linearly related to the time-invariant parameter vector sought, the simplifying independence setting assumptions facilitate deriving accurate closed-form expressions for the MSE steady-state values. The problems of mean- and MSE-sense stability of D-RLS are also investigated, and easily-checkable sufficient conditions are derived under which a steady-state is attained. Without resorting to diminishing step-sizes which compromise the tracking ability of D-RLS, stability ensures that per sensor estimates hover inside a ball of finite radius centered at the true parameter vector, with high-probability, even when inter-sensor communication links are noisy. Interestingly, computer simulations demonstrate that the theoretical findings are accurate also in the pragmatic settings whereby sensors acquire temporally-correlated data.

Index Terms

Wireless sensor networks (WSNs), distributed estimation, RLS algorithm, performance analysis.

EDICS Category: SEN-DIST, SPC-PERF, SEN-ASAL

[†] Work in this paper was supported by the NSF grants CCF-0830480 and ECCS-0824007.

* The authors are with the Dept. of Electrical and Computer Engineering, University of Minnesota, 200 Union Street SE, Minneapolis, MN 55455. Tel/fax: (612)626-7781/625-2002; Emails: {mate0058,georgios}@umn.edu

I. INTRODUCTION

Wireless sensor networks (WSNs), whereby large numbers of inexpensive sensors with constrained resources cooperate to achieve a common goal, constitute a promising technology for applications as diverse and crucial as environmental monitoring, process control and fault diagnosis for the industry, and protection of critical infrastructure including the smart grid, just to name a few. Emergent WSNs have created renewed interest also in the field of distributed computing, calling for collaborative solutions that enable low-cost estimation of stationary signals as well as reduced-complexity tracking of nonstationary processes; see e.g., [22], [33].

In this paper, a *distributed* recursive least-squares (D-RLS) algorithm is developed for estimation and tracking using ad hoc WSNs with noisy links, and analyzed in terms of its stability and mean-square error (MSE) steady-state performance. Ad hoc WSNs lack a central processing unit, and accordingly D-RLS performs in-network processing of the (spatially) distributed sensor observations. In words, a two-step iterative process takes place towards consenting on the desired global exponentially-weighted least-squares estimator (EWLSE): sensors perform simple local tasks to refine their current estimates, and exchange messages with one-hop neighbors over noisy communication channels. New sensor data acquired in real time enrich the estimation process and learn the unknown statistics ‘on-the-fly’. In addition, the exponential weighting effected through a forgetting factor endows D-RLS with tracking capabilities. This is desirable in a constantly changing environment, within which WSNs are envisioned to operate.

A. Prior art on distributed adaptive estimation

Unique challenges arising with WSNs dictate that often times sensors need to perform estimation in a constantly changing environment without having available a (statistical) model for the underlying processes of interest. This has motivated the development of *distributed adaptive* estimation schemes, generalizing the notion of adaptive filtering to a setup involving networked sensing/processing devices [3, Section I-B].

The incremental (I-) RLS algorithm in [24] is one of the first such approaches, which sequentially incorporates new sensor data while performing least-squares estimation. If one can afford maintaining a so-termed Hamiltonian cyclic path across sensors, then I-RLS yields the centralized EWLS benchmark estimate. Reducing the communication cost at a modest price in terms of estimation performance, an I-RLS variant was also put forth in [24]; but the NP-hard challenge of determining a Hamiltonian cycle in large-size WSNs remains [18]. Without topological constraints and increasing the degree of collaboration among sensors, a diffusion RLS algorithm was proposed in [3]. In addition to local estimates, sensors continuously diffuse raw sensor observations and regression vectors per neighborhood. This facilitates percolating new

data across the WSN, but estimation performance is degraded in the presence of communication noise. When both the sensor measurements and regression vectors are corrupted by additive (colored) noise, the diffusion-based RLS algorithm of [1] exploits sensor cooperation to reduce bias in the EWLSE. All [3], [1] and [24] include steady-state MSE performance analysis under the independence setting assumptions [23, p. 448]. Distributed least mean-squares (LMS) counterparts are also available, trading off computational complexity for estimation performance; for noteworthy representatives see [7], [14], [26], and references therein. Recent studies have also considered more elaborate sensor processing strategies including projections [8], [12], adaptive combination weights [30], or even sensor hierarchies [4], [26], and mobility [32].

Several distributed (adaptive) estimation algorithms are rooted on iterative optimization methods, which capitalize upon the separable structure of the cost defining the desired estimator. The sample mean estimator was formulated in [20] as an optimization problem, and was solved in a distributed fashion using a primal dual approach; see, e.g., [2]. Similarly, the incremental schemes in e.g., [7], [19], [21], [24] are all based on incremental (sub)gradient methods [17]. Even the diffusion LMS algorithm of [14] has been recently shown related to incremental strategies, when these are adopted to optimize an approximate reformulation of the LMS cost [5]. Building on the framework introduced by [27], the D-LMS and D-RLS algorithms in [15], [16], [26] are obtained upon recasting the respective decentralized estimation problems as multiple equivalent constrained subproblems. The resulting minimization subtasks are shown to be highly parallelizable across sensors, when carried out using the alternating-direction method of multipliers (AD-MoM) [2]. Much related to the AD-MoM is the alternating minimization algorithm (AMA) [31], used here to develop a novel D-RLS algorithm offering reduced complexity when compared to its counterpart of [15].

B. Contributions and paper outline

The present paper develops a fully distributed (D-) RLS type of algorithm, which performs in-network, adaptive LS estimation. D-RLS is applicable to general ad hoc WSNs that are challenged by additive communication noise, and may lack a Hamiltonian cycle altogether. Different from the distributed Kalman trackers of e.g., [6], [22], the universality of the LS principle broadens the applicability of D-RLS to a wide class of distributed adaptive estimation tasks, since it requires no knowledge of the underlying state space model. The algorithm is developed by reformulating the EWLSE into an equivalent constrained form [27], which can be minimized in a distributed fashion by capitalizing on the separable structure of the augmented Lagrangian using the AMA solver in [31] (Section II). From an algorithmic standpoint, the novel distributed iterations here offer two extra features relative to the AD-MoM-based D-RLS variants in [15], [25]. First, as discussed in Section II-B the per sensor computational complexity is markedly reduced, since there is no

need to explicitly carry out a matrix inversion per iteration as in [15]. Second, the approach here bypasses the need of the so-termed bridge sensors [25]. As a result, a fully distributed algorithm is obtained whereby all sensors perform the same tasks in a more efficient manner, without introducing hierarchies that may require intricate recovery protocols to cope with sensor failures.

Another contribution of the present paper pertains to a detailed stability and MSE steady-state *performance analysis* for D-RLS (Section IV). These theoretical results were lacking in the algorithmic papers [15], [25], where claims were only supported via computer simulations. Evaluating the performance of (centralized) adaptive filters is a challenging problem in its own right; prior art is surveyed in e.g., [28], [29, pg. 120], [23, pg. 357], and the extensive list of references therein. On top of that, a WSN setting introduces unique challenges in the analysis such as space-time sensor data and multiple sources of additive noise, a consequence of imperfect sensors and communication links. The approach pursued here capitalizes on an ‘averaged’ error-form representation of the local recursions comprising D-RLS, as a global dynamical system described by a stochastic difference-equation derived in Section III-B. The covariance matrix of the resulting state is then shown to encompass all the information needed to evaluate the relevant global and sensor-level performance metrics (Section III-C). For sensor observations that are linearly related to the time-invariant parameter vector sought, the simplifying independence setting assumptions [29, pg. 110], [23, pg. 448] are key enablers towards deriving accurate closed-form expressions for the mean-square deviation and excess-MSE steady-state values (Section IV-B). Stability in the mean- and MSE-sense are also investigated, revealing easily-checkable sufficient conditions under which a steady-state is attained.

Numerical tests corroborating the theoretical findings are presented in Section V, while concluding remarks and possible directions for future work are given in Section VI.

Notation: Operators \otimes , $(\cdot)^T$, $(\cdot)^\dagger$, $\lambda_{\max}(\cdot)$, $\text{tr}(\cdot)$, $\text{diag}(\cdot)$, $\text{bdiag}(\cdot)$, $E[\cdot]$, $\text{vec}[\cdot]$ will denote Kronecker product, transposition, matrix pseudo-inverse, spectral radius, matrix trace, diagonal matrix, block diagonal matrix, expectation, and matrix vectorization, respectively. For both vectors and matrices, $\|\cdot\|$ will stand for the 2–norm. and $|\cdot|$ for the cardinality of a set or the magnitude of a scalar. Positive definite matrix \mathbf{M} will be denoted by $\mathbf{M} \succ \mathbf{0}$. The $n \times n$ identity matrix will be represented by \mathbf{I}_n , while $\mathbf{1}_n$ will denote the $n \times 1$ vector of all ones and $\mathbf{1}_{n \times m} := \mathbf{1}_n \mathbf{1}_m^T$. Similar notation will be adopted for vectors (matrices) of all zeros. For matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$, $\text{nullspace}(\mathbf{M}) := \{\mathbf{x} \in \mathbb{R}^n : \mathbf{M}\mathbf{x} = \mathbf{0}_m\}$. The i -th vector in the canonical basis for \mathbb{R}^n will be denoted by $\mathbf{b}_{n,i}$, $i = 1, \dots, n$.

II. PROBLEM STATEMENT AND DISTRIBUTED RLS ALGORITHM

Consider a WSN with sensors $\{1, \dots, J\} := \mathcal{J}$. Only single-hop communications are allowed, i.e., sensor j can communicate only with the sensors in its neighborhood $\mathcal{N}_j \subseteq \mathcal{J}$, having cardinality $|\mathcal{N}_j|$. Assuming that inter-sensor links are symmetric, the WSN is modeled as an undirected connected graph with associated graph Laplacian matrix \mathbf{L} . Different from [1], [3] and [24], the present network model accounts explicitly for non-ideal sensor-to-sensor links. Specifically, signals received at sensor j from sensor i at discrete-time instant t are corrupted by a zero-mean additive noise vector $\boldsymbol{\eta}_j^i(t)$, assumed temporally and spatially uncorrelated. The communication noise covariance matrices are denoted by $\mathbf{R}_{\eta_j} := E[\boldsymbol{\eta}_j^i(t)(\boldsymbol{\eta}_j^i(t))^T]$, $j \in \mathcal{J}$.

The WSN is deployed to estimate a real signal vector $\mathbf{s}_0 \in \mathbb{R}^{p \times 1}$ in a distributed fashion and subject to the single-hop communication constraints, by resorting to the LS criterion [23, p. 658]. Per time instant $t = 0, 1, \dots$, each sensor acquires a regression vector $\mathbf{h}_j(t) \in \mathbb{R}^{p \times 1}$ and a scalar observation $x_j(t)$, both assumed zero-mean without loss of generality. A similar setting comprising complex-valued data was considered in [3] and [24]. Here, the exposition focuses on real-valued quantities for simplicity, but extensions to the complex case are straightforward. Given new data sequentially acquired, a pertinent approach is to consider the EWLSE [3], [23], [24]

$$\hat{\mathbf{s}}_{\text{ewls}}(t) := \arg \min_{\mathbf{s}} \sum_{\tau=0}^t \sum_{j=1}^J \lambda^{t-\tau} [x_j(\tau) - \mathbf{h}_j^T(\tau)\mathbf{s}]^2 + \lambda^t \mathbf{s}^T \boldsymbol{\Phi}_0 \mathbf{s} \quad (1)$$

where $\lambda \in (0, 1]$ is a forgetting factor, while $\boldsymbol{\Phi}_0 \succ \mathbf{0}_{p \times p}$ is included for regularization. Note that in forming the EWLSE at time t , the entire history of data $\{x_j(\tau), \mathbf{h}_j(\tau)\}_{\tau=0}^t, \forall j \in \mathcal{J}$ is incorporated in the online estimation process. Whenever $\lambda < 1$, past data are exponentially discarded thus enabling tracking of nonstationary processes. Regarding applications, a distributed power spectrum estimation task matching the aforementioned problem statement, can be found in [15].

To decompose the cost function in (1), in which summands are coupled through the global variable \mathbf{s} , introduce auxiliary variables $\{\mathbf{s}_j\}_{j=1}^J$ representing local estimates of \mathbf{s}_0 per sensor j . These local estimates are utilized to form the separable convex *constrained* minimization problem

$$\begin{aligned} \{\hat{\mathbf{s}}_j(t)\}_{j=1}^J &:= \arg \min_{\{\mathbf{s}_j\}_{j=1}^J} \sum_{\tau=0}^t \sum_{j=1}^J \lambda^{t-\tau} [x_j(\tau) - \mathbf{h}_j^T(\tau)\mathbf{s}_j]^2 + J^{-1} \lambda^t \sum_{j=1}^J \mathbf{s}_j^T \boldsymbol{\Phi}_0 \mathbf{s}_j, \\ \text{s. t. } \mathbf{s}_j &= \mathbf{s}_{j'}, \quad j \in \mathcal{J}, \quad j' \in \mathcal{N}_j. \end{aligned} \quad (2)$$

From the connectivity of the WSN, (1) and (2) are equivalent in the sense that $\hat{\mathbf{s}}_j(t) = \hat{\mathbf{s}}_{\text{ewls}}(t), \forall j \in \mathcal{J}$ and $t \geq 0$; see also [27]. To arrive at the D-RLS recursions, it is convenient to reparametrize the constraint

set (2) in the equivalent form

$$\mathbf{s}_j = \bar{\mathbf{z}}_j^{j'}, \quad \mathbf{s}_{j'} = \tilde{\mathbf{z}}_j^{j'}, \quad \text{and} \quad \bar{\mathbf{z}}_j^{j'} = \tilde{\mathbf{z}}_j^{j'}, \quad j \in \mathcal{J}, \quad j' \in \mathcal{N}_j, \quad j \neq j'. \quad (3)$$

where $\{\bar{\mathbf{z}}_j^{j'}, \tilde{\mathbf{z}}_j^{j'}\}_{j' \in \mathcal{N}_j, j \in \mathcal{J}}$, are auxiliary optimization variables that will be eventually eliminated.

A. The D-RLS algorithm

To tackle the constrained minimization problem (2) at time instant t , associate Lagrange multipliers $\mathbf{v}_j^{j'}$ and $\mathbf{u}_j^{j'}$ with the first pair of consensus constraints in (3). Introduce the ordinary Lagrangian function

$$\begin{aligned} \mathcal{L}[\mathbf{s}, \mathbf{z}, \mathbf{v}, \mathbf{u}] = & \sum_{j=1}^J \sum_{\tau=0}^t \lambda^{t-\tau} [x_j(\tau) - \mathbf{h}_j^T(\tau) \mathbf{s}_j]^2 + J^{-1} \lambda^t \sum_{j=1}^J \mathbf{s}_j^T \Phi_0 \mathbf{s}_j \\ & + \sum_{j=1}^J \sum_{j' \in \mathcal{N}_j} \left[(\mathbf{v}_j^{j'})^T (\mathbf{s}_j - \bar{\mathbf{z}}_j^{j'}) + (\mathbf{u}_j^{j'})^T (\mathbf{s}_{j'} - \tilde{\mathbf{z}}_j^{j'}) \right] \end{aligned} \quad (4)$$

as well as the quadratically *augmented* Lagrangian

$$\mathcal{L}_c[\mathbf{s}, \mathbf{z}, \mathbf{v}, \mathbf{u}] = \mathcal{L}[\mathbf{s}, \mathbf{z}, \mathbf{v}, \mathbf{u}] + \frac{c}{2} \sum_{j=1}^J \sum_{j' \in \mathcal{N}_j} \left[\|\mathbf{s}_j - \bar{\mathbf{z}}_j^{j'}\|_2^2 + \|\mathbf{s}_{j'} - \tilde{\mathbf{z}}_j^{j'}\|_2^2 \right] \quad (5)$$

where c is a positive penalty coefficient; and $\mathbf{s} := \{\mathbf{s}_j\}_{j=1}^J$, $\mathbf{z} := \{\bar{\mathbf{z}}_j^{j'}, \tilde{\mathbf{z}}_j^{j'}\}_{j \in \mathcal{J}, j' \in \mathcal{N}_j}$, and $[\mathbf{v}, \mathbf{u}] := \{\mathbf{v}_j^{j'}, \mathbf{u}_j^{j'}\}_{j \in \mathcal{J}, j' \in \mathcal{N}_j}$. Observe that the remaining constraints in (3), namely $\mathbf{z} \in C_z := \{\mathbf{z} : \bar{\mathbf{z}}_j^{j'} = \tilde{\mathbf{z}}_j^{j'}, j \in \mathcal{J}, j' \in \mathcal{N}_j, j \neq j'\}$, have not been dualized.

Towards deriving the D-RLS recursions, the alternating minimization algorithm (AMA) of [31] will be adopted here to tackle the separable EWLSE reformulation (2) in a distributed fashion. Much related to AMA is the alternating-direction method of multipliers (AD-MoM), an iterative augmented Lagrangian method specially well-suited for parallel processing [2], [15], [27]. While the AD-MoM has been proven successful to tackle the optimization tasks stemming from general distributed estimators of deterministic and (non-)stationary random signals, it is somehow curious that the AMA has remained largely underutilized.

To minimize (2) at time instant t , the AMA solver entails an iterative procedure comprising three steps per iteration $k = 0, 1, 2, \dots$

[S1] Multiplier updates:

$$\begin{aligned} \mathbf{v}_j^{j'}(t; k) &= \mathbf{v}_j^{j'}(t; k-1) + c[\mathbf{s}_j(t; k) - \bar{\mathbf{z}}_j^{j'}(t; k)], \quad j \in \mathcal{J}, j' \in \mathcal{N}_j \\ \mathbf{u}_j^{j'}(t; k) &= \mathbf{u}_j^{j'}(t; k-1) + c[\mathbf{s}_{j'}(t; k) - \tilde{\mathbf{z}}_j^{j'}(t; k)], \quad j \in \mathcal{J}, j' \in \mathcal{N}_j. \end{aligned}$$

[S2] Local estimate updates:

$$\mathbf{s}(t, k+1) = \arg \min_{\mathbf{s}} \mathcal{L}[\mathbf{s}, \mathbf{z}(t, k), \mathbf{v}(t, k), \mathbf{u}(t, k)]. \quad (6)$$

[S3] Auxiliary variable updates:

$$\mathbf{z}(t, k+1) = \arg \min_{\mathbf{z} \in C_z} \mathcal{L}_c[\mathbf{s}(t, k+1), \mathbf{z}, \mathbf{v}(t, k), \mathbf{u}(t, k)]. \quad (7)$$

Steps [S1] and [S3] are identical to those in AD-MoM [2]. In words, these steps correspond to dual ascent iterations to update the Lagrange multipliers, and a block coordinate-descent minimization of the augmented Lagrangian with respect to $\mathbf{z} \in C_z$, respectively. The only difference is with regards to the local estimate updates in [S2], where in AMA the new iterates are obtained by minimizing the ordinary Lagrangian with respect to \mathbf{s} . For the sake of the aforementioned minimization, all other variables are considered fixed taking their most up to date values $\{\mathbf{z}(t, k), \mathbf{v}(t, k), \mathbf{u}(t, k)\}$. For the AD-MoM instead, the minimized quantity is the augmented Lagrangian both in [S2] and in [S3].

The AMA was motivated in [31] for separable problems that are strictly convex in \mathbf{s} , but (possibly) only convex with respect to \mathbf{z} . Under this assumption, [S2] still yields a unique minimizer per iteration, and the AMA is useful for those cases in which the Lagrangian is much simpler to optimize than the augmented Lagrangian. Because of the regularization matrix $\Phi_0 \succ \mathbf{0}_{p \times p}$, the EWLS cost in (2) is indeed strictly convex for all $t > 0$, and the AMA is applicable. Section II-B dwells into the benefits of minimizing the ordinary Lagrangian instead of its augmented counterpart (5), in the context of distributed RLS estimation.

Carrying out the minimization in [S3] first, one finds

$$\bar{\mathbf{z}}_j^{j'}(t, k+1) = \tilde{\mathbf{z}}_j^{j'}(t, k+1) = \frac{1}{2} [\mathbf{s}_j(t, k+1) + \mathbf{s}_{j'}(t, k+1)], \quad j \in \mathcal{J}, j' \in \mathcal{N}_j$$

so that $\mathbf{v}_j^{j'}(t; k) = -\mathbf{u}_j^{j'}(t; k)$ for all $k > -1$ [15]. As a result $\mathbf{v}_j^{j'}(t; k)$ is given by

$$\mathbf{v}_j^{j'}(t; k) = \mathbf{v}_j^{j'}(t; k-1) + \frac{c}{2} [\mathbf{s}_j(t; k) - \mathbf{s}_{j'}(t; k)], \quad j \in \mathcal{J}, j' \in \mathcal{N}_j. \quad (8)$$

Moving on to [S2], from the separable structure of (4) the minimization (6) can be split into J subproblems

$$\mathbf{s}_j(t, k+1) = \arg \min_{\mathbf{s}_j} \left[\sum_{\tau=0}^t \lambda^{t-\tau} [x_j(\tau) - \mathbf{h}_j^T(\tau) \mathbf{s}_j]^2 + J^{-1} \lambda^t \mathbf{s}_j^T \Phi_0 \mathbf{s}_j + \sum_{j' \in \mathcal{N}_j} [\mathbf{v}_j^{j'}(t, k) - \mathbf{v}_{j'}^j(t, k)]^T \mathbf{s}_j \right].$$

Since each of the local subproblems corresponds to an unconstrained quadratic minimization, they all admit closed-form solutions

$$\mathbf{s}_j(t, k+1) = \Phi_j^{-1}(t) \boldsymbol{\psi}_j(t) - \frac{1}{2} \Phi_j^{-1}(t) \sum_{j' \in \mathcal{N}_j} [\mathbf{v}_j^{j'}(t, k) - \mathbf{v}_{j'}^j(t, k)] \quad (9)$$

where

$$\Phi_j(t) := \sum_{\tau=0}^t \lambda^{t-\tau} \mathbf{h}_j(\tau) \mathbf{h}_j^T(\tau) + J^{-1} \lambda^t \Phi_0 = \lambda \Phi_j(t-1) + \mathbf{h}_j(t) \mathbf{h}_j^T(t) \quad (10)$$

$$\boldsymbol{\psi}_j(t) := \sum_{\tau=0}^t \lambda^{t-\tau} \mathbf{h}_j(\tau) x_j(\tau) = \lambda \boldsymbol{\psi}_j(t-1) + \mathbf{h}_j(t) x_j(t). \quad (11)$$

Recursions (8) and (9) constitute the AMA-based D-RLS algorithm, whereby all sensors $j \in \mathcal{J}$ keep track of their local estimate $\mathbf{s}_j(t; k+1)$ and their multipliers $\{\mathbf{v}_j^{j'}(t; k)\}_{j' \in \mathcal{N}_j}$, which can be arbitrarily initialized. From the rank-one update in (10) and capitalizing on the matrix inversion lemma, matrix $\Phi_j^{-1}(t)$ can be efficiently updated according to

$$\Phi_j^{-1}(t) = \lambda^{-1} \Phi_j^{-1}(t-1) - \frac{\lambda^{-1} \Phi_j^{-1}(t) \mathbf{h}_j(t) \mathbf{h}_j^T(t) \Phi_j^{-1}(t-1)}{\lambda + \mathbf{h}_j^T(t) \Phi_j^{-1}(t-1) \mathbf{h}_j(t)}. \quad (12)$$

with complexity $\mathcal{O}(p^2)$. It is recommended to initialize the matrix recursion with $\Phi_j^{-1}(0) = J \Phi_0^{-1} := \delta \mathbf{I}_p$, where $\delta > 0$ is chosen sufficiently large [23]. Not surprisingly, by direct application of the convergence results in [31, Proposition 3], it follows that:

Proposition 1: *For arbitrarily initialized $\{\mathbf{v}_j^{j'}(t; -1)\}_{j' \in \mathcal{N}_j}^{j' \in \mathcal{N}_j}$, $\mathbf{s}_j(t; 0)$ and $c \in (0, c_u)$; the local estimates $\mathbf{s}_j(t; k)$ generated by (9) reach consensus as $k \rightarrow \infty$; i.e.,*

$$\lim_{k \rightarrow \infty} \mathbf{s}_j(t; k) = \hat{\mathbf{s}}_{\text{ewls}}(t), \text{ for all } j \in \mathcal{J}.$$

The upper bound c_u is proportional to the modulus of the strictly convex cost function in (2), and inversely proportional to the norm of a matrix suitably chosen to express the linear constraints in (3); further details are in [31, Section 4]. Proposition 1 asserts that per time instant t , the AMA-based D-RLS algorithm yields a sequence of local estimates that converge to the global EWLSE sought, as $k \rightarrow \infty$, or, pragmatically for large enough k . In principle, one could argue that running many consensus iterations may not be a problem in a stationary environment. However, when the WSN is deployed to track a time-varying parameter vector $\mathbf{s}_0(t)$, one cannot afford significant delays in-between consecutive sensing instants.

One possible way to overcome this hurdle is to run a single consensus iteration per acquired observation $x_j(t)$. Specifically, letting $k = t$ in recursions (8)-(9), one arrives at a single time scale D-RLS algorithm which is suitable for operation in nonstationary WSN environments. Accounting also for additive communication noise that corrupts the exchanges of multipliers and local estimates, the per sensor tasks comprising the novel AMA-based *single time scale* D-RLS algorithm are given by

$$\mathbf{v}_j^{j'}(t) = \mathbf{v}_j^{j'}(t-1) + \frac{c}{2} \left[\mathbf{s}_j(t) - (\mathbf{s}_{j'}(t) + \boldsymbol{\eta}_j^{j'}(t)) \right], \quad j' \in \mathcal{N}_j \quad (13)$$

$$\Phi_j^{-1}(t+1) = \lambda^{-1} \Phi_j^{-1}(t) - \frac{\lambda^{-1} \Phi_j^{-1}(t) \mathbf{h}_j(t+1) \mathbf{h}_j^T(t+1) \Phi_j^{-1}(t)}{\lambda + \mathbf{h}_j^T(t+1) \Phi_j^{-1}(t) \mathbf{h}_j(t+1)} \quad (14)$$

$$\boldsymbol{\psi}_j(t+1) = \lambda \boldsymbol{\psi}_j(t) + \mathbf{h}_j(t+1) x_j(t+1) \quad (15)$$

$$\mathbf{s}_j(t+1) = \Phi_j^{-1}(t+1) \boldsymbol{\psi}_j(t+1) - \frac{1}{2} \Phi_j^{-1}(t+1) \sum_{j' \in \mathcal{N}_j} \left[\mathbf{v}_j^{j'}(t) - (\mathbf{v}_{j'}^j(t) + \bar{\boldsymbol{\eta}}_j^{j'}(t)) \right]. \quad (16)$$

Algorithm 1 : AMA-based D-RLS

Arbitrarily initialize $\{\mathbf{s}_j(0)\}_{j=1}^J$ and $\{\mathbf{v}_j^{j'}(-1)\}_{j \in \mathcal{J}}^{j' \in \mathcal{N}_j}$.

for $t = 0, 1, \dots$ **do**

 All $j \in \mathcal{J}$: transmit $\mathbf{s}_j(t)$ to neighbors in \mathcal{N}_j .

 All $j \in \mathcal{J}$: update $\{\mathbf{v}_j^{j'}(t)\}_{j' \in \mathcal{N}_j}$ using (13).

 All $j \in \mathcal{J}$: transmit $\mathbf{v}_j^{j'}(t)$ to each $j' \in \mathcal{N}_j$.

 All $j \in \mathcal{J}$: update $\bar{\Phi}_j(t+1)$ and $\psi_j(t+1)$ using (14) and (15), respectively.

 All $j \in \mathcal{J}$: update $\mathbf{s}_j(t+1)$ using (16).

end for

Recursions (13)-(15) are tabulated as Algorithm 1, which also details the inter-sensor communications of multipliers and local estimates taking place within neighborhoods. When powerful error control codes render inter-sensor links virtually ideal, direct application of the results in [15], [16] show that D-RLS can be further simplified to reduce the communication overhead and memory storage requirements.

B. Comparison with the AD-MoM-based D-RLS algorithm

A related D-RLS algorithm was put forth in [15], whereby the decomposable exponentially-weighted LS cost (2) is minimized using the AD-MoM, rather than the AMA as in Section II-A. Recall that the AD-MoM solver yields $\mathbf{s}_j(t+1)$ as the optimizer of the augmented Lagrangian, while its AMA counterpart minimizes the ordinary Lagrangian instead. Consequently, different from (16) local estimates in the AD-MoM-based D-RLS algorithm of [15] are updated via

$$\begin{aligned} \mathbf{s}_j(t+1) = & \bar{\Phi}_j^{-1}(t+1)\psi_j(t+1) + \frac{c}{2}\bar{\Phi}_j^{-1}(t+1) \sum_{j' \in \mathcal{N}_j} \left[\mathbf{s}_j(t) + (\mathbf{s}_{j'}(t) + \bar{\boldsymbol{\eta}}_j^{j'}(t)) \right] \\ & - \frac{1}{2}\bar{\Phi}_j^{-1}(t+1) \sum_{j' \in \mathcal{N}_j} \left[\mathbf{v}_j^{j'}(t) - (\mathbf{v}_{j'}^j(t) + \bar{\boldsymbol{\eta}}_j^{j'}(t)) \right] \end{aligned} \quad (17)$$

where [cf. (10)]

$$\bar{\Phi}_j(t) := \sum_{\tau=0}^t \lambda^{t-\tau} \mathbf{h}_j(\tau) \mathbf{h}_j^T(\tau) + J^{-1} \lambda^t \bar{\Phi}_0 + c|\mathcal{N}_j| \mathbf{I}_p. \quad (18)$$

Unless $\lambda = 1$, it is impossible to derive a rank-one update for $\bar{\Phi}_j(t)$ as in (10). The reason is the regularization term $c|\mathcal{N}_j| \mathbf{I}_p$ in (18), a direct consequence of the quadratic penalty in the augmented Lagrangian (5). This prevents one from efficiently updating $\bar{\Phi}_j^{-1}(t+1)$ in (17) using the matrix inversion lemma [cf. (14)]. Direct inversion of $\bar{\Phi}_j(t+1)$ per iteration dominates the computational complexity of the AD-MoM-based D-RLS algorithm, which is roughly $\mathcal{O}(p^3)$ [15].

Unfortunately, the penalty coefficient cannot be set to zero because the D-RLS algorithm breaks down. For instance, when the initial Lagrange multipliers are null and $c = 0$, D-RLS boils down to a purely local (L-) RLS algorithm where sensors do not cooperate, hence consensus cannot be attained. All in all, the novel AMA-based D-RLS algorithm of this paper offers an improved alternative with an order of magnitude reduction in terms of computational complexity per sensor. With regards to communication cost, the AD-MoM-based D-RLS and Algorithm 1 here incur identical overheads; see [15, Sec. III-B] for a detailed analysis of the associated cost, as well as comparisons with the I-RLS [24] and diffusion RLS algorithms [3].

While the AMA-based D-RLS algorithm is less complex computationally than its AD-MoM counterpart in [15], Proposition 1 asserts that when many consensus iterations can be afforded, convergence to the centralized EWLSE is guaranteed provided $c \in (0, c_u)$. On the other hand, the AD-MoM-based D-RLS algorithm will attain the EWLSE for any $c > 0$ (cf. [15, Prop. 1]). In addition, it does not require tuning the extra parameter δ , since it is applicable when $\Phi_0 = \mathbf{0}_{p \times p}$ because the augmented Lagrangian provides the needed regularization.

III. ANALYSIS PRELIMINARIES

A. Scope of the analysis: assumptions and approximations

Performance evaluation of the D-RLS algorithm is much more involved than that of e.g., D-LMS [16], [26]. The challenges are well documented for the classical (centralized) LMS and RLS filters [23], [29], and results for the latter are less common and typically involve simplifying approximations. What is more, the distributed setting introduces unique challenges in the analysis. These include space-time sensor data and multiple sources of additive noise, a consequence of imperfect sensors and communication links.

In order to proceed, a few typical modeling assumptions are introduced to delineate the scope of the ensuing stability and performance results. For all $j \in \mathcal{J}$, it is assumed that:

- (a1) *Sensor observations adhere to the linear model $x_j(t) = \mathbf{h}_j^T(t)\mathbf{s}_0 + \epsilon_j(t)$, where the zero-mean white noise $\{\epsilon_j(t)\}$ has variance $\sigma_{\epsilon_j}^2$;*
- (a2) *Vectors $\{\mathbf{h}_j(t)\}$ are spatio-temporally white with covariance matrix $\mathbf{R}_{h_j} \succ \mathbf{0}_{p \times p}$; and*
- (a3) *Vectors $\{\mathbf{h}_j(t)\}$, $\{\epsilon_j(t)\}$, $\{\boldsymbol{\eta}_j^{j'}(t)\}_{j' \in \mathcal{N}_j}$ and $\{\bar{\boldsymbol{\eta}}_j^{j'}(t)\}_{j' \in \mathcal{N}_j}$ are independent.*

Assumptions (a1)-(a3) comprise the widely adopted *independence setting*, for sensor observations that are linearly related to the time-invariant parameter of interest; see e.g., [29, pg. 110], [23, pg. 448]. Clearly, (a2) can be violated in, e.g., FIR filtering of signals (regressors) with a shift structure as in the distributed power spectrum estimation problem described in [26] and [15]. Nevertheless, the steady-state performance

results extend accurately to the pragmatic setup that involves time-correlated sensor data; see also the numerical tests in Section V. In line with a distributed setting such as a WSN, the statistical profiles of both regressors and the noise quantities vary across sensors (space), yet they are assumed to remain time invariant. For a related analysis of a distributed LMS algorithm operating in a nonstationary environment, the reader is referred to [16].

In the particular case of the D-RLS algorithm, a unique challenge stems from the stochastic matrices $\Phi_j^{-1}(t)$ present in the local estimate updates (16). Recalling (10), it is apparent that $\Phi_j^{-1}(t)$ depends upon the *whole history* of local regression vectors $\{\mathbf{h}_j(\tau)\}_{\tau=0}^t$. Even obtaining $\Phi_j^{-1}(t)$'s distribution or computing its expected value is a formidable task in general, due to the matrix inversion operation. It is for these reasons that some simplifying approximations will be adopted in the sequel, to carry out the analysis that otherwise becomes intractable.

Neglecting the regularization term in (10) that vanishes exponentially as $t \rightarrow \infty$, the matrix $\Phi_j(t)$ is obtained as an exponentially weighted moving average (EWMA). The EWMA can be seen as an average modulated by a sliding window of equivalent length $1/(1 - \lambda)$, which clearly grows as $\lambda \rightarrow 1$. This observation in conjunction with (a2) and the strong law of large numbers, justifies the approximation

$$\Phi_j(t) \approx E[\Phi_j(t)] = \frac{\mathbf{R}_{h_j}}{1 - \lambda}, \quad 0 \ll \lambda < 1 \text{ and } t \rightarrow \infty. \quad (19)$$

The expectation of $\Phi_j^{-1}(t)$, on the other hand, is considerably harder to evaluate. To overcome this challenge, the following approximation will be invoked [3], [23]

$$E[\Phi_j^{-1}(t)] \approx E^{-1}[\Phi_j(t)] \approx (1 - \lambda)\mathbf{R}_{h_j}^{-1}, \quad 0 \ll \lambda < 1 \text{ and } t \rightarrow \infty. \quad (20)$$

It is admittedly a crude approximation at first sight, because $E[X^{-1}] \neq E[X]^{-1}$ in general, for any random variable X . However, experimental evidence suggests that the approximation is sufficiently accurate for all practical purposes, when the forgetting factor approaches unity [23, p. 319].

B. Error-form D-RLS

The approach here to steady-state performance analysis relies on an ‘averaged’ error-form system representation of D-RLS in (13)-(16), where $\Phi_j^{-1}(t)$ in (16) is replaced by the approximation $(1 - \lambda)\mathbf{R}_{h_j}^{-1}$, for sufficiently large t . Somehow related approaches were adopted in [3] and [1]. Other noteworthy analysis techniques include the energy-conservation methodology in [35], [23, p. 287], and stochastic averaging [29, p. 229]. For performance analysis of distributed adaptive algorithms seeking time-invariant parameters, the former has been applied in e.g., [13], [14], while the latter can be found in [26].

Towards obtaining such error-form representation, introduce the local estimation errors $\{\mathbf{y}_{1,j}(t) := \mathbf{s}_j(t) - \mathbf{s}_0\}_{j=1}^J$ and multiplier-based quantities $\{\mathbf{y}_{2,j}(t) := \frac{1}{2} \sum_{j' \in \mathcal{N}_j} (\mathbf{v}_{j'}^{j'}(t-1) - \mathbf{v}_{j'}^j(t-1))\}_{j=1}^J$. It turns out that a convenient global state to describe the spatio-temporal dynamics of D-RLS in (13)-(16) is $\mathbf{y}(t) := [\mathbf{y}_1^T(t) \ \mathbf{y}_2^T(t)]^T = [\mathbf{y}_{1,1}^T(t) \dots \mathbf{y}_{1,J}^T(t) \ \mathbf{y}_{2,1}^T(t) \dots \mathbf{y}_{2,J}^T(t)]^T \in \mathbb{R}^{2Jp}$. In addition, to concisely capture the effects of both observation and communication noise on the estimation errors across the WSN, define the $Jp \times 1$ noise supervectors $\boldsymbol{\epsilon}(t) := \sum_{\tau=0}^t \lambda^{t-\tau} [\mathbf{h}_1^T(\tau) \epsilon_1(\tau) \dots \mathbf{h}_J^T(\tau) \epsilon_J(\tau)]^T$ and $\bar{\boldsymbol{\eta}}(t) := [\bar{\boldsymbol{\eta}}_1^T(t) \dots \bar{\boldsymbol{\eta}}_J^T(t)]^T$. Vectors $\{\bar{\boldsymbol{\eta}}_j(t)\}_{j=1}^J$ represent the aggregate noise corrupting the multipliers received by sensor j at time instant t , and are given by

$$\bar{\boldsymbol{\eta}}_j(t) := \frac{1}{2} \sum_{j' \in \mathcal{N}_j} \bar{\boldsymbol{\eta}}_j^{j'}(t). \quad (21)$$

Their respective covariance matrices are easily computable under (a2)-(a3). For instance,

$$\mathbf{R}_{\boldsymbol{\epsilon}}(t) := E[\boldsymbol{\epsilon}(t)\boldsymbol{\epsilon}^T(t)] = \frac{1 - \lambda^{2(t+1)}}{1 - \lambda^2} \text{bdiag}(\mathbf{R}_{h_1} \sigma_{\epsilon_1}^2, \dots, \mathbf{R}_{h_J} \sigma_{\epsilon_J}^2) \quad (22)$$

while the structure of $\mathbf{R}_{\bar{\boldsymbol{\eta}}} := E[\bar{\boldsymbol{\eta}}(t)\bar{\boldsymbol{\eta}}^T(t)]$ is given in Appendix E. Two additional $Jp \times 1$ communication noise supervectors are needed, namely $\boldsymbol{\eta}_\alpha(t) := [(\boldsymbol{\eta}_1^\alpha(t))^T \dots (\boldsymbol{\eta}_J^\alpha(t))^T]^T$ and $\boldsymbol{\eta}_\beta(t) := [(\boldsymbol{\eta}_1^\beta(t))^T \dots (\boldsymbol{\eta}_J^\beta(t))^T]^T$, where for $j \in \mathcal{J}$

$$\boldsymbol{\eta}_j^\alpha(t) := \frac{c}{4} \sum_{j' \in \mathcal{N}_j} \boldsymbol{\eta}_j^{j'}(t), \quad \boldsymbol{\eta}_j^\beta(t) := \frac{c}{4} \sum_{j' \in \mathcal{N}_j} \boldsymbol{\eta}_j^j(t). \quad (23)$$

Finally, let $(c/2)\mathbf{L} \otimes \mathbf{I}_p \in \mathbb{R}^{Jp \times Jp}$ be a matrix capturing the WSN connectivity pattern through the (scaled) graph Laplacian matrix \mathbf{L} , and define $\mathbf{R}_h^{-1} := \text{bdiag}(\mathbf{R}_{h_1}^{-1}, \dots, \mathbf{R}_{h_J}^{-1})$. Based on these definitions, it is possible to state the following important lemma established in Appendix A.

Lemma 1: *Let (a1) and (a2) hold. Then for $t \geq t_0$ with t_0 sufficiently large while $0 \ll \lambda < 1$, the global state $\mathbf{y}(t)$ approximately evolves according to*

$$\begin{aligned} \mathbf{y}(t+1) = \text{bdiag}((1-\lambda)\mathbf{R}_h^{-1}, \mathbf{I}_{Jp}) \left\{ \boldsymbol{\Upsilon} \mathbf{y}(t) + \begin{bmatrix} \mathbf{I}_{Jp} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \boldsymbol{\epsilon}(t+1) + \begin{bmatrix} \mathbf{I}_{Jp} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t) \right. \\ \left. + \begin{bmatrix} \mathbf{I}_{Jp} \\ -\mathbf{I}_{Jp} \end{bmatrix} \boldsymbol{\eta}_\alpha(t) - \begin{bmatrix} \mathbf{I}_{Jp} \\ -\mathbf{I}_{Jp} \end{bmatrix} \boldsymbol{\eta}_\beta(t) \right\} \end{aligned} \quad (24)$$

where the $2Jp \times 2Jp$ matrix $\boldsymbol{\Upsilon}$ consists of the $Jp \times Jp$ blocks $[\boldsymbol{\Upsilon}]_{11} = -[\boldsymbol{\Upsilon}]_{21} = -\mathbf{L}_c$ and $[\boldsymbol{\Upsilon}]_{12} = -[\boldsymbol{\Upsilon}]_{22} = -\mathbf{I}_{Jp}$. The initial condition $\mathbf{y}(t_0)$ should be selected as $\mathbf{y}(t_0) = \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \mathbf{y}'(t_0)$, where $\mathbf{y}'(t_0)$ is any vector in \mathbb{R}^{2Jp} .

The convenience of representing $\mathbf{y}(t)$ as in Lemma 1 will become apparent in the sequel, especially when investigating sufficient conditions under which the D-RLS algorithm is stable in the mean sense

(Section IV-A). In addition, the covariance matrix of the state vector $\mathbf{y}(t)$ can be shown to encompass all the information needed to evaluate the relevant per sensor and networkwide performance figures of merit, the subject dealt with next.

C. Performance Metrics

When it comes to performance evaluation of adaptive algorithms, it is customary to consider as figures of merit the so-called MSE, excess mean-square error (EMSE), and mean-square deviation (MSD) [23], [29]. In the present setup for distributed adaptive estimation, it is pertinent to address both global (networkwide) and local (per-sensor) performance [14]. After recalling the definitions of the local a priori error $e_j(t) := x_j(t) - \mathbf{h}_j^T(t)\mathbf{s}_j(t-1)$ and local estimation error $\mathbf{y}_{1,j}(t) := \mathbf{s}_j(t) - \mathbf{s}_0$, the per-sensor performance metrics are defined as

$$\text{MSE}_j(t) := E[e_j^2(t)]$$

$$\text{EMSE}_j(t) := E[(\mathbf{h}_j^T(t)\mathbf{y}_{1,j}(t-1))^2]$$

$$\text{MSD}_j(t) := E[\|\mathbf{y}_{1,j}(t)\|^2]$$

whereas their global counterparts are defined as the respective averages across sensors, e.g., $\text{MSE}(t) := J^{-1} \sum_{j=1}^J E[e_j(t)^2]$, and so on.

Next, it is shown that it suffices to evaluate the state covariance matrix $\mathbf{R}_y(t) := E[\mathbf{y}(t)\mathbf{y}^T(t)]$ in order to assess the aforementioned performance metrics. To this end, note that by virtue of (a1) it is possible to write $e_j(t) = -\mathbf{h}_j^T(t)\mathbf{y}_{1,j}(t-1) + \epsilon_j(t)$. Because $\mathbf{y}_{1,j}(t-1)$ is independent of the zero-mean $\{\mathbf{h}_j(t), \epsilon_j(t)\}$ under (a1)-(a3), from the previous relationship between the a priori and estimation errors one finds that $\text{MSE}_j(t) = \text{EMSE}_j(t) + \sigma_{\epsilon_j}^2$. Hence, it suffices to focus on the evaluation of $\text{EMSE}_j(t)$, through which $\text{MSE}_j(t)$ can also be determined under the assumption that the observation noise variances are known, or can be estimated for that matter. If $\mathbf{R}_{y_{1,j}}(t) := E[\mathbf{y}_{1,j}(t)\mathbf{y}_{1,j}^T(t)]$ denotes the j -th local error covariance matrix, then $\text{MSD}_j(t) = \text{tr}(\mathbf{R}_{y_{1,j}}(t))$; and under (a1)-(a3), a simple manipulation yields

$$\begin{aligned} \text{EMSE}_j(t) &= E[\text{tr}((\mathbf{h}_j^T(t)\mathbf{y}_{1,j}(t-1))^2)] = \text{tr}(E[\mathbf{h}_j(t)\mathbf{h}_j^T(t)\mathbf{y}_{1,j}(t-1)\mathbf{y}_{1,j}^T(t-1)]) \\ &= \text{tr}(E[\mathbf{h}_j(t)\mathbf{h}_j^T(t)]E[\mathbf{y}_{1,j}(t-1)\mathbf{y}_{1,j}^T(t-1)]) = \text{tr}(\mathbf{R}_{h_j}\mathbf{R}_{y_{1,j}}(t-1)). \end{aligned}$$

To derive corresponding formulas for the global performance figures of merit, let $\mathbf{R}_{y_1}(t) := E[\mathbf{y}_1(t)\mathbf{y}_1^T(t)]$ denote the global error covariance matrix, and define $\mathbf{R}_h := E[\mathbf{R}_h(t)] = \text{bdiag}(\mathbf{R}_{h_1}, \dots, \mathbf{R}_{h_J})$. It follows that $\text{MSD}(t) = J^{-1}\text{tr}(\mathbf{R}_{y_1}(t))$, and $\text{EMSE}(t) = J^{-1}\text{tr}(\mathbf{R}_h\mathbf{R}_{y_1}(t-1))$.

TABLE I
EVALUATION OF LOCAL AND GLOBAL FIGURES OF MERIT FROM $\mathbf{R}_y(t)$

	MSD	EMSE	MSE
Local	$\text{tr}([\mathbf{R}_y(t)]_{11,j})$	$\text{tr}(\mathbf{R}_{h_j}[\mathbf{R}_y(t-1)]_{11,j})$	$\text{tr}(\mathbf{R}_{h_j}[\mathbf{R}_y(t-1)]_{11,j}) + \sigma_{\epsilon_j}^2$
Global	$J^{-1}\text{tr}([\mathbf{R}_y(t)]_{11})$	$J^{-1}\text{tr}(\mathbf{R}_h[\mathbf{R}_y(t-1)]_{11})$	$J^{-1}\text{tr}(\mathbf{R}_h[\mathbf{R}_y(t-1)]_{11}) + J^{-1}\sum_{j=1}^J \sigma_{\epsilon_j}^2$

It is now straightforward to recognize that $\mathbf{R}_y(t)$ indeed provides all the information needed to evaluate the performance of the D-RLS algorithm. For instance, observe that the global error covariance matrix $\mathbf{R}_{y_1}(t)$ corresponds to the $Jp \times Jp$ upper left submatrix of $\mathbf{R}_y(t)$, which is denoted by $[\mathbf{R}_y(t)]_{11}$. Further, the j -th $p \times p$ diagonal submatrix ($j = 1, \dots, J$) of $[\mathbf{R}_y(t)]_{11}$ is exactly $\mathbf{R}_{y_{1,j}}(t)$, and is likewise denoted by $[\mathbf{R}_y(t)]_{11,j}$. For clarity, the aforementioned notational conventions regarding submatrices within $\mathbf{R}_y(t)$ are illustrated in Fig. 1. In a nutshell, deriving a closed-form expression for $\mathbf{R}_y(t)$ enables the evaluation of all performance metrics of interest, as summarized in Table I. This task will be considered in Section IV-B.

Remark 1 Since the ‘average’ system representation of $\mathbf{y}(t)$ in (24) relies on an approximation that becomes increasingly accurate as $\lambda \rightarrow 1$ and $t \rightarrow \infty$, so does the covariance recursion for $\mathbf{R}_y(t)$ derived in Section IV-B. For this reason, the scope of the MSE performance analysis of this paper pertains to the *steady-state* behavior of the D-RLS algorithm.

IV. STABILITY AND STEADY-STATE PERFORMANCE ANALYSIS

In this section, stability and steady-state performance analyses are conducted for the D-RLS algorithm developed in Section II-A. Because recursions (13)-(16) are stochastic in nature, stability will be assessed both in the mean- and in the MSE-sense. The techniques presented here can be utilized with minimal modifications to derive analogous results for the AD-MoM-based D-RLS algorithm in [15].

A. Mean Stability

Based on Lemma 1, it follows that D-RLS achieves consensus in the mean sense on the parameter \mathbf{s}_0 .

Proposition 2: Under (a1)-(a3) and for $0 \ll \lambda < 1$, D-RLS achieves consensus in the mean, i.e.,

$$\lim_{t \rightarrow \infty} E[\mathbf{y}_{1,j}(t)] = \mathbf{0}_p, \quad \forall j \in \mathcal{J}$$

provided the penalty coefficient is chosen such that

$$0 < c < \frac{4}{(1-\lambda)\lambda_{\max}(\mathbf{R}_h^{-1}(\mathbf{L} \otimes \mathbf{I}_p))}. \quad (25)$$

Proof: Based on (a1)-(a3) and since the data is zero-mean, one obtains after taking expectations on (24) that $E[\mathbf{y}(t)] = \text{bdiag}((1-\lambda)\mathbf{R}_h^{-1}, \mathbf{I}_{Jp})\Upsilon E[\mathbf{y}(t-1)]$. The following lemma characterizes the spectrum of the transition matrix $\Omega := \text{bdiag}((1-\lambda)\mathbf{R}_h^{-1}, \mathbf{I}_{Jp})\Upsilon$; see Appendix B for a proof.

Lemma 2: *Regardless of the value of $c > 0$, matrix $\Omega := \text{bdiag}((1-\lambda)\mathbf{R}_h^{-1}, \mathbf{I}_{Jp})\Upsilon \in \mathbb{R}^{2Jp \times 2Jp}$ has p eigenvalues equal to one. Further, the left eigenvectors associated with the unity eigenvalue have the structure $\mathbf{v}_i^T = [\mathbf{0}_{1 \times Jp} \quad \mathbf{q}_i^T]$, where $\mathbf{q}_i \in \text{nullspace}(\mathbf{L}_c)$ and $i = 1, \dots, p$. The remaining eigenvalues are equal to zero, or else have modulus strictly smaller than one provided c satisfies the bound (25).*

Back to establishing the mean stability result, let $\{\mathbf{u}_i\}$ and $\{\mathbf{v}_i^T\}$ respectively denote the collection of p right and left eigenvectors of Ω associated with the eigenvalue one. By virtue of Lemma 2 and provided c satisfies the bound (25), one has that $\lim_{t \rightarrow \infty} \Omega^t = \sum_{i=1}^p \mathbf{u}_i \mathbf{v}_i^T$; hence,

$$\begin{aligned} \lim_{t \rightarrow \infty} E[\mathbf{y}(t)] &= \left(\sum_{i=1}^p \mathbf{u}_i \mathbf{v}_i^T \right) \mathbf{y}(t_0) = \left(\sum_{i=1}^p \mathbf{u}_i \mathbf{v}_i^T \right) \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \mathbf{y}'(t_0) \\ &= \left(\sum_{i=1}^p \mathbf{u}_i [\mathbf{0}_{1 \times Jp} \quad \mathbf{q}_i^T \mathbf{L}_c] \right) \mathbf{y}'(t_0) = \mathbf{0}_{2Jp}. \end{aligned}$$

In obtaining the second equality, the structure for $\mathbf{y}(t_0)$ that is given in Lemma 1 was used. The last equality follows from the fact that $\mathbf{q}_i \in \text{nullspace}(\mathbf{L}_c)$ as per Lemma 2, thus completing the proof. ■

Before wrapping up this section, a comment is due on the sufficient condition (25). When performing distributed estimation under $0 \ll \lambda < 1$, the condition is actually not restrictive at all since a $1-\lambda$ factor is present in the denominator. When λ is close to one, any practical choice of $c > 0$ will result in asymptotically unbiased sensor estimates. Also note that the bound depends on the WSN topology, through the scaled graph Laplacian matrix \mathbf{L}_c .

B. MSE Stability and Steady-State Performance

In order to assess the steady-state MSE performance of the D-RLS algorithm, we will evaluate the figures of merit introduced in Section III-C. The limiting values of both the local (per sensor) and global (network-wide) MSE, excess mean-square error (EMSE), and mean-square deviation (MSD), will be assessed. To this end, it suffices to derive a closed-form expression for the global estimation error covariance matrix $\mathbf{R}_{y_1}(t) := E[\mathbf{y}_1(t)\mathbf{y}_1^T(t)]$, as already argued in Section III-C.

The next result provides an equivalent representation of the approximate D-RLS global recursion (24), that is more suitable for the recursive evaluation of $\mathbf{R}_{y_1}(t)$. First, introduce the $p(\sum_{j=1}^J |\mathcal{N}_j|) \times 1$ vector

$$\boldsymbol{\eta}(t) := [\{(\boldsymbol{\eta}_{j'}^1(t))^T\}_{j' \in \mathcal{N}_1} \dots \{(\boldsymbol{\eta}_{j'}^J(t))^T\}_{j' \in \mathcal{N}_J}]^T \quad (26)$$

which comprises the receiver noise terms corrupting transmissions of local estimates across the whole network at time instant t , and define $\mathbf{R}_{\boldsymbol{\eta}} := E[\boldsymbol{\eta}(t)\boldsymbol{\eta}^T(t)]$. For notational convenience, let $\mathbf{R}_{h,\lambda}^{-1} := (1 - \lambda)\mathbf{R}_h^{-1}$.

Lemma 3: *Under the assumptions of Lemma 1, the global state $\mathbf{y}(t)$ in (24) can be equivalently written as*

$$\mathbf{y}(t+1) = \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c)\mathbf{z}(t+1) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{P}_\beta - \mathbf{P}_\alpha \end{bmatrix} \boldsymbol{\eta}(t). \quad (27)$$

The inner state $\mathbf{z}(t) := [\mathbf{z}_1^T(t) \mathbf{z}_2^T(t)]^T$ is arbitrarily initialized at time t_0 , and updated according to

$$\mathbf{z}(t+1) = \boldsymbol{\Psi}\mathbf{z}(t) + \boldsymbol{\Psi} \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix} \boldsymbol{\eta}(t-1) + \boldsymbol{\Psi} \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t-1) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \boldsymbol{\epsilon}(t+1) \quad (28)$$

where the $2Jp \times 2Jp$ transition matrix $\boldsymbol{\Psi}$ consists of the blocks $[\boldsymbol{\Psi}]_{11} = [\boldsymbol{\Psi}]_{12} = -\mathbf{R}_{h,\lambda}^{-1}\mathbf{L}_c$ and $[\boldsymbol{\Psi}]_{21} = [\boldsymbol{\Psi}]_{22} = \mathbf{L}_c\mathbf{L}_c^\dagger$. Matrix \mathbf{C} is chosen such that $\mathbf{L}_c\mathbf{C} = \mathbf{P}_\beta - \mathbf{P}_\alpha$, where the structure of the time-invariant matrices \mathbf{P}_α and \mathbf{P}_β is given in Appendix E.

Proof: See Appendix C. ■

The desired state $\mathbf{y}(t)$ is obtained as a rank-deficient linear transformation of the inner state $\mathbf{z}(t)$, plus a stochastic offset due to the presence of communication noise. A linear, time-invariant, first-order difference equation describes the dynamics of $\mathbf{z}(t)$, and hence of $\mathbf{y}(t)$, via the algebraic transformation in (27). The time-invariant nature of the transition matrix $\boldsymbol{\Psi}$ is due to the approximations $\Phi_j^{-1}(t) \approx \mathbf{R}_{h,\lambda}^{-1}$, $j \in \mathcal{J}$, particularly accurate for large enough $t > t_0$. Examination of (28) reveals that the evolution of $\mathbf{z}(t)$ is driven by three stochastic input processes: i) communication noise $\boldsymbol{\eta}(t-1)$ affecting the transmission of local estimates; ii) communication noise $\bar{\boldsymbol{\eta}}(t-1)$ contaminating the Lagrange multipliers; and iii) observation noise within $\boldsymbol{\epsilon}(t+1)$.

Focusing now on the calculation of $\mathbf{R}_{y_1}(t) = [\mathbf{R}_y(t)]_{11}$ based on Lemma 3, observe from the upper $Jp \times 1$ block of $\mathbf{y}(t+1)$ in (27) that $\mathbf{y}_1(t+1) = \mathbf{z}_1(t+1) + \mathbf{R}_{h,\lambda}^{-1}[\bar{\boldsymbol{\eta}}(t) + (\mathbf{P}_\alpha - \mathbf{P}_\beta)\boldsymbol{\eta}(t)]$. Under (a3), $\mathbf{z}_1(t+1)$ is independent of the zero-mean $\{\bar{\boldsymbol{\eta}}(t), \boldsymbol{\eta}(t)\}$; hence,

$$\mathbf{R}_{y_1}(t) = \mathbf{R}_{z_1}(t) + \mathbf{R}_{h,\lambda}^{-1} [\mathbf{R}_{\bar{\boldsymbol{\eta}}} + (\mathbf{P}_\alpha - \mathbf{P}_\beta)\mathbf{R}_{\boldsymbol{\eta}}(\mathbf{P}_\alpha - \mathbf{P}_\beta)^T] \mathbf{R}_{h,\lambda}^{-1} \quad (29)$$

which prompts one to obtain $\mathbf{R}_z(t) := E[\mathbf{z}(t)\mathbf{z}^T(t)]$. Specifically, the goal is to extract its upper-left $Jp \times Jp$ matrix block $[\mathbf{R}_z(t)]_{11} = \mathbf{R}_{z_1}(t)$. To this end, define the vectors

$$\bar{\boldsymbol{\eta}}_\lambda(t) := \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t), \quad \boldsymbol{\eta}_\lambda(t) := \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix} \boldsymbol{\eta}(t) \quad (30)$$

whose respective covariance matrices $\mathbf{R}_{\bar{\boldsymbol{\eta}}_\lambda} := E[\bar{\boldsymbol{\eta}}_\lambda(t)\bar{\boldsymbol{\eta}}_\lambda^T(t)]$ and $\mathbf{R}_{\boldsymbol{\eta}_\lambda} := E[\boldsymbol{\eta}_\lambda(t)\boldsymbol{\eta}_\lambda^T(t)]$ have a structure detailed in Appendix E. Also recall that $\boldsymbol{\epsilon}(t)$ depends on the entire history of regressors up to time instant t . Starting from (28) and capitalizing on (a2)-(a3), it is straightforward to obtain a first-order matrix recursion to update $\mathbf{R}_z(t)$ as

$$\begin{aligned} \mathbf{R}_z(t) &= \boldsymbol{\Psi}\mathbf{R}_z(t-1)\boldsymbol{\Psi}^T + \boldsymbol{\Psi}\mathbf{R}_{\bar{\boldsymbol{\eta}}_\lambda}\boldsymbol{\Psi}^T + \boldsymbol{\Psi}\mathbf{R}_{\boldsymbol{\eta}_\lambda}\boldsymbol{\Psi}^T + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \mathbf{R}_\epsilon(t) \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix}^T \\ &\quad + \boldsymbol{\Psi}\mathbf{R}_{z\epsilon}(t) \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix}^T + \left(\boldsymbol{\Psi}\mathbf{R}_{z\epsilon}(t) \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix}^T \right)^T \end{aligned} \quad (31)$$

$$:= \boldsymbol{\Psi}\mathbf{R}_z(t-1)\boldsymbol{\Psi}^T + \mathbf{R}_\nu(t) \quad (32)$$

where the cross-correlation matrix $\mathbf{R}_{z\epsilon}(t) := E[\mathbf{z}(t-1)\boldsymbol{\epsilon}^T(t)]$ is recursively updated as (cf. Appendix D)

$$\mathbf{R}_{z\epsilon}(t) = \lambda \boldsymbol{\Psi}\mathbf{R}_{z\epsilon}(t-1) + \lambda \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \mathbf{R}_\epsilon(t-1). \quad (33)$$

For notational brevity in what follows, $\mathbf{R}_\nu(t)$ in (32) denotes all the covariance forcing terms in the right-hand side of (31). The main result of this section pertains to MSE stability of the D-RLS algorithm, and provides a checkable sufficient condition under which the global error covariance matrix $\mathbf{R}_{y_1}(t)$ has bounded entries as $t \rightarrow \infty$. Recall that a matrix is termed stable, when all its eigenvalues lie strictly inside the unit circle.

Proposition 3: *Under (a1)-(a3) and for $0 \ll \lambda < 1$, D-RLS is MSE stable, i.e., $\lim_{t \rightarrow \infty} \mathbf{R}_{y_1}(t)$ has bounded entries, provided that $c > 0$ is chosen so that $\boldsymbol{\Psi}$ is a stable matrix.*

Proof: First observe that because $\lambda \in (0, 1)$, it holds that

$$\begin{aligned} \lim_{t \rightarrow \infty} \mathbf{R}_\epsilon(t) &= \lim_{t \rightarrow \infty} \left(\frac{1 - \lambda^{2(t+1)}}{1 - \lambda^2} \right) \text{bdiag}(\mathbf{R}_{h_1}\sigma_{\epsilon_1}^2, \dots, \mathbf{R}_{h_J}\sigma_{\epsilon_J}^2) \\ &= \left(\frac{1}{1 - \lambda^2} \right) \text{bdiag}(\mathbf{R}_{h_1}\sigma_{\epsilon_1}^2, \dots, \mathbf{R}_{h_J}\sigma_{\epsilon_J}^2) =: \mathbf{R}_\epsilon(\infty). \end{aligned} \quad (34)$$

If $c > 0$ is selected such that Ψ is a stable matrix, then clearly $\lambda\Psi$ is also stable, and hence the matrix recursion (33) converges to the bounded limit

$$\mathbf{R}_{z\epsilon}(\infty) = (\mathbf{I}_{2Jp} - \lambda\Psi)^{-1} \begin{bmatrix} \lambda\mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \mathbf{R}_{\epsilon}(\infty). \quad (35)$$

Based on the previous arguments, it follows that the forcing matrix $\mathbf{R}_{\nu}(t)$ in (31) will also attain a bounded limit as $t \rightarrow \infty$, denoted as $\mathbf{R}_{\nu}(\infty)$. Next, we show that $\lim_{t \rightarrow \infty} \mathbf{R}_z(t)$ has bounded entries by studying its equivalent vectorized dynamical system. Upon vectorizing (32), it follows that

$$\begin{aligned} \text{vec}[\mathbf{R}_z(t)] &= \text{vec}[\Psi\mathbf{R}_z(t-1)\Psi^T] + \text{vec}[\mathbf{R}_{\nu}(t)] \\ &= (\Psi \otimes \Psi) \text{vec}[\mathbf{R}_z(t-1)] + \text{vec}[\mathbf{R}_{\nu}(t)] \end{aligned}$$

where in obtaining the last equality we used the property $\text{vec}[\mathbf{RST}] = (\mathbf{T}^T \otimes \mathbf{R}) \text{vec}[\mathbf{S}]$. Because the eigenvalues of $\Psi \otimes \Psi$ are the pairwise products of those of Ψ , stability of Ψ implies stability of the Kronecker product. As a result, the vectorized recursion will converge to the limit

$$\text{vec}[\mathbf{R}_z(\infty)] = (\mathbf{I}_{(2Jp)^2} - \Psi \otimes \Psi)^{-1} \text{vec}[\mathbf{R}_{\nu}(\infty)] \quad (36)$$

which of course implies that $\lim_{t \rightarrow \infty} \mathbf{R}_z(t) = \mathbf{R}_z(\infty)$ has bounded entries. From (29), the same holds true for $\mathbf{R}_{y_1}(t)$, and the proof is completed. \blacksquare

Proposition 3 asserts that the AMA-based D-RLS algorithm is stable in the MSE-sense, even when the WSN links are challenged by additive noise. While most distributed adaptive estimation works have only looked at ideal inter-sensor links, others have adopted diminishing step-sizes to mitigate the undesirable effects of communication noise [10], [11]. This approach however, limits their applicability to stationary environments. Remarkably, the AMA-based D-RLS algorithm exhibits robustness to noise when using a constant step-size c , a feature that has also been observed for AD-MoM related distributed iterations in e.g., [26], [27], and [15].

As a byproduct, the proof of Proposition 3 also provides part of the recipe towards evaluating the steady-state MSE performance of the D-RLS algorithm. Indeed, by plugging (34) and (35) into (31) one obtains the steady-state covariance matrix $\mathbf{R}_{\nu}(\infty)$. It is then possible to evaluate $\mathbf{R}_z(\infty)$, by reshaping the vectorized identity (36). Matrix $\mathbf{R}_{z_1}(\infty)$ can be extracted from the upper-left $Jp \times Jp$ matrix block of $\mathbf{R}_z(\infty)$, and the desired global error covariance matrix $\mathbf{R}_{y_1}(\infty) = [\mathbf{R}_y(\infty)]_{11}$ becomes available via (29). Closed-form evaluation of the $\text{MSE}(\infty)$, $\text{EMSE}(\infty)$ and $\text{MSD}(\infty)$ for every sensor $j \in \mathcal{J}$ is now possible given $\mathbf{R}_{y_1}(\infty)$, by resorting to the formulae in Table I.

Before closing this section, an alternative notion of stochastic stability that readily follows from Proposition 3 is established here. Specifically, it is possible to show that under the independence setting assumptions

(a1)-(a3) considered so far, the global error norm $\|\mathbf{y}_1(t)\|$ remains most of the time within a finite interval, i.e., errors are weakly stochastic bounded (WSB) [28], [29, pg. 110]. This WSB stability guarantees that for any $\theta > 0$, there exists a $\zeta > 0$ such that $\Pr[\|\mathbf{y}_1(t)\| < \zeta] = 1 - \theta$ uniformly in time.

Corollary 1: *Under (a1)-(a3) and for $0 \ll \lambda < 1$, if $c > 0$ is chosen so that Ψ is a stable matrix, then the D-RLS algorithm yields estimation errors which are WSB; i.e., $\lim_{\zeta \rightarrow \infty} \sup_{t \geq t_0} \Pr[\|\mathbf{y}_1(t)\| \geq \zeta] = 0$.*

Proof: Chebyshev's inequality implies that

$$\Pr[\|\mathbf{y}_1(t)\| \geq \zeta] \leq \frac{E[\|\mathbf{y}_1(t)\|^2]}{\zeta^2} = \frac{\text{tr}([\mathbf{R}_y(t)]_{11})}{\zeta^2}. \quad (37)$$

From Proposition 3, $\lim_{t \rightarrow \infty} [\mathbf{R}_y(t)]_{11}$ has bounded entries, implying that $\sup_{t \geq t_0} \text{tr}([\mathbf{R}_y(t)]_{11}) < \infty$. Taking the limit as $\zeta \rightarrow \infty$, while relying on the bound in (37) which holds for all values of $t \geq t_0$, yields the desired result. ■

In words, Corollary 1 ensures that with overwhelming probability, local sensor estimates remain inside a ball with finite radius, centered at \mathbf{s}_0 . It is certainly a weak notion of stability, many times the only one that can be asserted when the presence of, e.g., time-correlated data, renders variance calculations impossible; see also [26], [28]. In this case where stronger assumptions are invoked, WSB follows immediately once MSE-sense stability is established. Nevertheless, it is an important practical notion as it ensures – on a per-realization basis – that estimation errors have no probability mass escaping to infinity. In particular, D-RLS estimation errors are shown WSB in the presence of communication noise; a property not enjoyed by other distributed iterations for e.g., consenting on averages [34].

V. NUMERICAL TESTS

Computer simulations are carried out here to corroborate the analytical results of Section IV-B. Even though based on simplifying assumptions and approximations, the usefulness of the analysis is justified since the predicted steady-state MSE figures of merit accurately match the empirical D-RLS limiting values. In accordance with the adaptive filtering folklore, when $\lambda \rightarrow 1$ the upshot of the analysis under the independence setting assumptions is shown to extend accurately to the pragmatic scenario whereby sensors acquire time-correlated data. For $J = 15$ sensors, a connected ad hoc WSN is generated as a realization of the random geometric graph model on the unit-square, with communication range $r = 0.3$ [9]. To model non-ideal inter-sensor links, additive white Gaussian noise (AWGN) with variance $\sigma_\eta^2 = 10^{-1}$ is added at the receiving end. The WSN used for the experiments is depicted in Fig. 2.

With $p = 4$ and $\mathbf{s}_0 = \mathbf{1}_p$, observations obey a linear model [cf. (a1)] with sensing WGN of spatial variance profile $\sigma_{\epsilon_j}^2 = 10^{-3}\alpha_j$, where $\alpha_j \sim \mathcal{U}[0, 1]$ (uniform distribution) and i.i.d.. The regression vectors

$\mathbf{h}_j(t) := [h_j(t) \dots h_j(t - p + 1)]^T$ have a shift structure, and entries which evolve according to first-order stable autoregressive processes $h_j(t) = (1 - \rho)\beta_j h_j(t - 1) + \sqrt{\rho}\omega_j(t)$ for all $j \in \mathcal{J}$. We choose $\rho = 5 \times 10^{-1}$, the $\beta_j \sim \mathcal{U}[0, 1]$ i.i.d. in space, and the driving white noise $\omega_j(t) \sim \mathcal{U}[-\sqrt{3}\sigma_{\omega_j}, \sqrt{3}\sigma_{\omega_j}]$ with spatial variance profile given by $\sigma_{\omega_j}^2 = 2\gamma_j$ with $\gamma_j \sim \mathcal{U}[0, 1]$ and i.i.d.. Observe that the data is temporally-correlated, implying that (a2) does not hold here.

For all experimental performance curves obtained by running the algorithms, the ensemble averages are approximated by sample averaging 200 runs of the experiment.

First, with $\lambda = 0.95$, $c = 0.1$ and $\delta = 100$ for the AMA-based D-RLS algorithm, Fig. 3 depicts the network performance through the evolution of the EMSE(t) and MSD(t) figures of merit. Both noisy and ideal links are considered. The steady-state limiting values found in Section IV-B are extremely accurate, even though the simulated data does not adhere to (a2), and the results are based on simplifying approximations. As intuitively expected and analytically corroborated via the noise-related additive terms in (29) and (31), the performance penalty due to non-ideal links is also apparent.

We also utilize the analytical results developed throughout this paper to contrast the per sensor performance of D-RLS and the D-LMS algorithm in [16]. In particular, the parameters chosen for D-LMS are $\mu = 5 \times 10^{-3}$ and $c = 1$. Fig. 4 shows the values of the EMSE $_j(\infty)$ and MSD $_j(\infty)$ for all $j \in \mathcal{J}$. As expected, the second-order D-RLS scheme attains improved steady-state performance uniformly across all sensors in the simulated WSN. In this particular simulated test, gains as high as 5dB in estimation error can be achieved at the price of increasing computational burden per sensor, from $\mathcal{O}(p)$ to $\mathcal{O}(p^2)$ per iteration.

VI. CONCLUDING SUMMARY AND FUTURE WORK

A distributed RLS-like algorithm is developed in this paper, which is capable of performing adaptive estimation and tracking using WSNs in which sensors cooperate with single-hop neighbors. The WSNs considered here are quite general since they do not necessarily possess a Hamiltonian cycle, while the inter-sensor links are challenged by communication noise. Distributed iterations are derived after: i) reformulating in a separable way the exponentially weighed least-squares (EWLS) cost involved in the classical RLS algorithm; and ii) applying the AMA to minimize this separable cost in a distributed fashion. The AMA is especially well-suited to capitalize on the strict convexity of the EWLS cost, and thus offer significant reductions in computational complexity per sensor, when compared to existing alternatives. This way, salient features of the classical RLS algorithm are shown to carry over to a distributed WSN setting, namely reduced-complexity estimation when a state and/or data model is not available and fast convergence rates are at a premium.

An additional contribution of this paper pertains to a detailed steady-state MSE performance analysis, that relies on an ‘averaged’ error-form system representation of D-RLS. The theory is developed under some simplifying approximations, and resorting to the independence setting assumptions. This way, it is possible to obtain accurate closed-form expressions for both the per sensor and network-wide relevant performance metrics as $t \rightarrow \infty$. Sufficient conditions under which the D-RLS algorithm is stable in the mean- and MSE-sense are provided as well. As a corollary, the D-RLS estimation errors are also shown to remain within a finite interval with high probability, even when the inter-sensor links are challenged by additive noise. Numerical simulations demonstrated that the analytical findings of this paper extend accurately to a more realistic WSN setting, whereby sensors acquire temporally correlated sensor data.

Regarding the performance of the D-RLS algorithm, there are still several interesting directions to pursue as future work. First, it would be nice to establish a stochastic *trajectory locking* result which formally shows that as $\lambda \rightarrow 1$, the D-RLS estimation error trajectories closely follow the ones of its time-invariant ‘averaged’ system companion. Second, the steady-state MSE performance analysis was carried out when $0 \ll \lambda < 1$. For the infinite memory case in which $\lambda = 1$, numerical simulations indicate that D-RLS provides mean-square sense-consistent estimates, even in the presence of communication noise. By formally establishing this property, D-RLS becomes an even more appealing alternative for distributed parameter estimation in stationary environments. While the approximations used in this paper are no longer valid when $\lambda = 1$, for Gaussian i.i.d. regressors matrix $\Phi^{-1}(t)$ is Wishart distributed with known moments. Under these assumptions, consistency analysis is a subject of ongoing investigation.

APPENDIX

A. Proof of Lemma 1: Let t_0 be chosen large enough to ensure that

$$\lim_{t \rightarrow t_0} \Phi_j(t) = \lim_{t \rightarrow t_0} \sum_{\tau=0}^t \lambda^{t-\tau} \mathbf{h}_j(\tau) \mathbf{h}_j^T(\tau) + J^{-1} \lambda^t \Phi_0 \approx \frac{\mathbf{R}_{h_j}}{1-\lambda}, \quad j \in \mathcal{J}.$$

For $t > t_0$, consider replacing $\Phi_j^{-1}(t)$ in (16) with the approximation $(1-\lambda)\mathbf{R}_{h_j}^{-1}$ for its expected value, to arrive at the ‘average’ D-RLS system recursions

$$\mathbf{v}_j^{j'}(t) = \mathbf{v}_j^{j'}(t-1) + \frac{c}{2} \left[\mathbf{s}_j(t) - (\mathbf{s}_{j'}(t) + \boldsymbol{\eta}_j^{j'}(t)) \right], \quad j' \in \mathcal{N}_j \quad (38)$$

$$\mathbf{s}_j(t+1) = (1-\lambda)\mathbf{R}_{h_j}^{-1} \boldsymbol{\psi}_j(t+1) - \frac{1}{2}(1-\lambda)\mathbf{R}_{h_j}^{-1} \sum_{j' \in \mathcal{N}_j} \left[\mathbf{v}_j^{j'}(t) - (\mathbf{v}_{j'}^j(t) + \bar{\boldsymbol{\eta}}_j^{j'}(t)) \right] \quad (39)$$

After summing $(\mathbf{v}_{j'}^{j'}(t) - \mathbf{v}_{j'}^j(t))/2$ over $j' \in \mathcal{N}_j$, it follows from (38) that for all $j \in \mathcal{J}$

$$\mathbf{y}_{2,j}(t+1) := \frac{1}{2} \sum_{j' \in \mathcal{N}_j} (\mathbf{v}_{j'}^{j'}(t) - \mathbf{v}_{j'}^j(t)) = \mathbf{y}_{2,j}(t) + \frac{c}{2} \sum_{j' \in \mathcal{N}_j} (\mathbf{s}_j(t) - \mathbf{s}_{j'}(t)) - \frac{c}{4} \sum_{j' \in \mathcal{N}_j} (\boldsymbol{\eta}_{j'}^{j'}(t) - \boldsymbol{\eta}_{j'}^j(t)) \quad (40)$$

$$= \mathbf{y}_{2,j}(t) + \frac{c}{2} \sum_{j' \in \mathcal{N}_j} (\mathbf{y}_{1,j}(t) - \mathbf{y}_{1,j'}(t)) - \boldsymbol{\eta}_j^\alpha(t) + \boldsymbol{\eta}_j^\beta(t), \quad (41)$$

where the last equality was obtained after adding and subtracting $c|\mathcal{N}_j|\mathbf{s}_0$ from the right-hand side of (40), and relying on the definitions in (23). Next, starting from (39) and upon: i) using (a1) to eliminate

$$\psi_j(t+1) = \sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} \mathbf{h}_j(\tau) \mathbf{h}_j^T(\tau) \mathbf{s}_0 + \sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} \mathbf{h}_j(\tau) \epsilon_j(\tau) \approx \frac{\mathbf{R}_{h_j}}{1-\lambda} \mathbf{s}_0 + \sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} \mathbf{h}_j(\tau) \epsilon_j(\tau)$$

from (39); ii) recognizing $\mathbf{y}_{2,j}(t+1)$ in the right-hand side of (39) and substituting it with (41); and iii) replacing the sums of noise vectors with the quantities defined in (21) and (23); one arrives at

$$\begin{aligned} \mathbf{y}_{1,j}(t+1) &= (1-\lambda) \mathbf{R}_{h_j}^{-1} \left[-\frac{c}{2} \sum_{j' \in \mathcal{N}_j} (\mathbf{y}_{1,j}(t) - \mathbf{y}_{1,j'}(t)) - \mathbf{y}_{2,j}(t) \right] \\ &\quad + (1-\lambda) \mathbf{R}_{h_j}^{-1} \left[\sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} \mathbf{h}_j(\tau) \epsilon_j(\tau) + \boldsymbol{\eta}_j^\alpha(t) - \boldsymbol{\eta}_j^\beta(t) + \bar{\boldsymbol{\eta}}_j(t) \right]. \end{aligned} \quad (42)$$

What remains to be shown is that after stacking the recursions (42) and (41) for $j = 1, \dots, J$ to form the one for $\mathbf{y}(t+1)$, we can obtain the compact representation in (24). Examining (41) and (42), it is apparent that a common matrix factor $\text{bdiag}((1-\lambda)\mathbf{R}_{h_j}^{-1}, \mathbf{I}_{Jp})$ can be pulled out to simplify the expression for $\mathbf{y}(t+1)$. Consider first the forcing terms in (24). Stacking the channel noise terms from (42) and (41), readily yields the last three terms inside the curly brackets in (24). Likewise, stacking the terms $\sum_{\tau=0}^{t+1} \lambda^{t+1-\tau} \mathbf{h}_j(\tau) \epsilon_j(\tau)$ for $j = 1, \dots, J$ yields the second term due to the observation noise; recall the definition of $\boldsymbol{\epsilon}(t+1)$. This term as well as the vectors $\bar{\boldsymbol{\eta}}_j(t)$ are not present in (41), which explains the zero vector at the lower part of the second and third terms inside the curly brackets of (24).

To specify the structure of the transition matrix $\boldsymbol{\Upsilon}$, note that the first term on the right-hand side of (41) explains why $[\boldsymbol{\Upsilon}]_{22} = \mathbf{I}_{Jp}$. Similarly, the second term inside the first square brackets in (42) explains why $[\boldsymbol{\Upsilon}]_{12} = -\mathbf{I}_{Jp}$. Next, it follows readily that upon stacking the terms $(c/2) \sum_{j' \in \mathcal{N}_j} (\mathbf{y}_{1,j}(t) - \mathbf{y}_{1,j'}(t))$, which correspond to a scaled Laplacian-based combination of $p \times 1$ vectors, one obtains $[(c/2)\mathbf{L} \otimes \mathbf{I}_p] \mathbf{y}_1(t) = \mathbf{L}_c \mathbf{y}_1(t)$. This justifies why $[\boldsymbol{\Upsilon}]_{11} = -[\boldsymbol{\Upsilon}]_{21} = -\mathbf{L}_c$.

A comment is due regarding the initialization for $t = t_0$. Although the vectors $\{\mathbf{y}_{1,j}(t_0)\}_{j=1}^J$ are decoupled so that $\mathbf{y}_1(t_0)$ can be chosen arbitrarily, this is not the case for $\{\mathbf{y}_{2,j}(t_0)\}_{j=1}^J$ which are coupled and satisfy

$$\sum_{j=1}^J \mathbf{y}_{2,j}(t) = \sum_{j=1}^J \sum_{j' \in \mathcal{N}_j} (\mathbf{v}_{j'}^{j'}(t-1) - \mathbf{v}_{j'}^j(t-1)) = \mathbf{0}_p, \quad \forall t \geq 0. \quad (43)$$

The coupling across $\{\mathbf{y}_{2,j}(t)\}_{j=1}^J$ dictates $\mathbf{y}_2(t_0)$ to be chosen in compliance with (43), so that the system (24) is equivalent to (38) and (39) for all $t \geq t_0$. Let $\mathbf{y}_2(t_0) = \mathbf{L}_c \mathbf{y}'_2(t_0)$, where $\mathbf{y}'_2(t_0)$ is any vector in \mathbb{R}^{Jp} . Then, it is not difficult to see that $\mathbf{y}_2(t_0)$ satisfies the conservation law (43). In conclusion, for arbitrary $\mathbf{y}'(0) \in \mathbb{R}^{2Jp}$ the recursion (24) should be initialized as $\mathbf{y}(0) = \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \mathbf{y}'(0)$, and the proof of Lemma 1 is completed. \blacksquare

B. Proof of Lemma 2: Recall the structure of matrix Υ given in Lemma 1. A vector $\mathbf{v}_i^T := [\mathbf{v}_{1,i}^T \ \mathbf{v}_{2,i}^T]$ with $\{\mathbf{v}_{j,i}\}_{j=1}^2 \in \mathbb{R}^{Jp \times 1}$ is a left eigenvector of Ω associated to the eigenvalue one, if and only if it solves the following linear system of equations

$$\begin{aligned} -\mathbf{v}_{1,i}^T(1-\lambda)\mathbf{R}_h^{-1}\mathbf{L}_c + \mathbf{v}_{2,i}^T\mathbf{L}_c &= \mathbf{v}_{1,i}^T \\ -\mathbf{v}_{1,i}^T(1-\lambda)\mathbf{R}_h^{-1} + \mathbf{v}_{2,i}^T &= \mathbf{v}_{2,i}^T \end{aligned}$$

The second equation can only be satisfied for $\mathbf{v}_{1,i} = \mathbf{0}_{Jp}$, and upon substituting this value in the first equation one obtains that $\mathbf{v}_{2,i} \in \text{nullspace}(\mathbf{L}_c) = \text{nullspace}(\mathbf{L} \otimes \mathbf{I}_p)$ for all values of $c > 0$. Under the assumption of a connected ad hoc WSN, $\text{nullspace}(\mathbf{L}) = \text{span}(\mathbf{1}_J)$ and hence $\text{nullspace}(\mathbf{L} \otimes \mathbf{I}_p)$ is a p -dimensional subspace.

Following steps similar to those in [27, Appendix H], it is possible to express the eigenvalues of Ω that are different from one as the roots of a second-order polynomial. Such a polynomial does not have an independent term, so that some eigenvalues are zero. With respect to the rest of the eigenvalues, it is possible to show that their magnitude is upper bounded by $\lambda_{\max}(\mathbf{I}_{Jp} - (1-\lambda)\mathbf{R}_h^{-1}\mathbf{L}_c)$. Hence, it is possible to select $c > 0$ such that $\lambda_{\max}(\mathbf{I}_{Jp} - (1-\lambda)\mathbf{R}_h^{-1}\mathbf{L}_c) < 1$, or equivalently $|1 - (1-\lambda)\lambda_{\max}(\mathbf{R}_h^{-1}\mathbf{L}_c)| < 1$, which is the same as condition (25). \blacksquare

C. Proof of Lemma 3: The goal is to establish the equivalence between the dynamical systems in (24) and (27) for all $t \geq t_0$, when the inner state is arbitrarily initialized as $\mathbf{z}(t_0) = \mathbf{y}'(t_0)$. We will argue by induction. For $t = t_0$, it follows from (28) that $\mathbf{z}(t_0 + 1) = \Psi \mathbf{y}'(t_0) + [\mathbf{R}_{h,\lambda}^{-1} \ \mathbf{0}^T]^T \boldsymbol{\epsilon}(t_0 + 1)$, since (by convention) there is no communication noise for $t < t_0$. Upon substituting $\mathbf{z}(t_0 + 1)$ into (27), we find

$$\mathbf{y}(t_0 + 1) = \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \Psi \mathbf{y}'(t_0) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} (\boldsymbol{\epsilon}(t_0 + 1) + \bar{\boldsymbol{\eta}}(t_0)) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{P}_\beta - \mathbf{P}_\alpha \end{bmatrix} \boldsymbol{\eta}(t_0). \quad (44)$$

Note that: i) $\text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \Psi = \Upsilon \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c)$; ii) $\mathbf{y}(t_0) = \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \mathbf{y}'(t_0)$ for the system in Lemma 1; and iii) $\boldsymbol{\eta}_\alpha(t) = \mathbf{P}_\alpha \boldsymbol{\eta}(t)$, while $\boldsymbol{\eta}_\beta(t) = \mathbf{P}_\beta \boldsymbol{\eta}(t)$ [cf. Appendix E]. Thus, the right-hand side of (44) is equal to the right-hand side of (24) for $t = t_0$.

Suppose next that (27) and (28) hold true for $\mathbf{y}(t)$ and $\mathbf{z}(t)$, with $t \geq t_0$. The same will be shown for $\mathbf{y}(t+1)$ and $\mathbf{z}(t+1)$. To this end, replace $\mathbf{y}(t)$ with the right-hand side of (27) evaluated at time t , into

(24) to obtain

$$\begin{aligned}
\mathbf{y}(t+1) &= \text{bdiag}(\mathbf{R}_{h,\lambda}^{-1}, \mathbf{I}_{Jp}) \left\{ \Upsilon \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \mathbf{z}(t) + \Upsilon \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t-1) + \begin{bmatrix} \mathbf{I}_{Jp} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \boldsymbol{\epsilon}(t+1) \right. \\
&\quad \left. + \Upsilon \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{P}_\beta - \mathbf{P}_\alpha \end{bmatrix} \boldsymbol{\eta}(t-1) + \begin{bmatrix} \mathbf{I}_{Jp} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t) + \begin{bmatrix} \mathbf{I}_{Jp} \\ -\mathbf{I}_{Jp} \end{bmatrix} \boldsymbol{\eta}_\alpha(t) - \begin{bmatrix} \mathbf{I}_{Jp} \\ -\mathbf{I}_{Jp} \end{bmatrix} \boldsymbol{\eta}_\beta(t) \right\} \\
&= \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \left(\Psi \mathbf{z}(t) + \Psi \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t-1) + \Psi \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix} \boldsymbol{\eta}(t-1) \right. \\
&\quad \left. + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \boldsymbol{\epsilon}(t+1) \right) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \bar{\boldsymbol{\eta}}(t) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{P}_\beta - \mathbf{P}_\alpha \end{bmatrix} \boldsymbol{\eta}(t) \quad (45)
\end{aligned}$$

where in obtaining the last equality in (45), the following were used: i) $\text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c) \Psi = \Upsilon \text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c)$; ii) the relationship between $\boldsymbol{\eta}_\alpha(t)$, $\boldsymbol{\eta}_\beta(t)$ and $\boldsymbol{\eta}(t)$ given in Appendix E; and iii) the existence of a matrix \mathbf{C} such that $\mathbf{L}_c \mathbf{C} = \mathbf{P}_\beta - \mathbf{P}_\alpha$. This made possible to extract the common factor $\text{bdiag}(\mathbf{I}_{Jp}, \mathbf{L}_c)$ and deduce from (45) that $\mathbf{y}(t+1)$ is given by (27), while $\mathbf{z}(t+1)$ is provided by (28).

In order to complete the proof, one must show the existence of matrix \mathbf{C} . To this end, via a simple evaluation one can check that $\text{nullspace}(\mathbf{L}_c) \subseteq \text{nullspace}(\mathbf{P}_\beta^T - \mathbf{P}_\alpha^T)$, and since \mathbf{L}_c is symmetric, one has $\text{nullspace}(\mathbf{L}_c) \perp \text{range}(\mathbf{L}_c)$. As $\text{nullspace}(\mathbf{P}_\beta^T - \mathbf{P}_\alpha^T) \perp \text{range}(\mathbf{P}_\beta - \mathbf{P}_\alpha)$, it follows that $\text{range}(\mathbf{P}_\beta - \mathbf{P}_\alpha) \subseteq \text{range}(\mathbf{L}_c)$, which further implies that there exists \mathbf{C} such that $\mathbf{L}_c \mathbf{C} = \mathbf{P}_\beta - \mathbf{P}_\alpha$. ■

D. Derivation of (33): First observe that the noise supervector $\boldsymbol{\epsilon}(t)$ obeys the first-order recursion

$$\boldsymbol{\epsilon}(t) := \sum_{\tau=0}^t \lambda^{t-\tau} [\mathbf{h}_1^T(\tau) \epsilon_1(\tau) \dots \mathbf{h}_J^T(\tau) \epsilon_J(\tau)]^T = \lambda \boldsymbol{\epsilon}(t-1) + [\mathbf{h}_1^T(t) \epsilon_1(t) \dots \mathbf{h}_J^T(t) \epsilon_J(t)]^T. \quad (46)$$

Because under (a3) the zero-mean $\{\epsilon_j(t)\}_{j \in \mathcal{J}}$ are independent of $\mathbf{z}(t-1)$ [cf. (28)], it follows readily that $\mathbf{R}_z \boldsymbol{\epsilon}(t) := E[\mathbf{z}(t-1) \boldsymbol{\epsilon}^T(t)] = \lambda E[\mathbf{z}(t-1) \boldsymbol{\epsilon}^T(t-1)]$. Plugging the expression for $\mathbf{z}(t-1)$ and carrying out the expectation yields

$$\begin{aligned}
E[\mathbf{z}(t-1) \boldsymbol{\epsilon}^T(t-1)] &= \Psi E[\mathbf{z}(t-2) \boldsymbol{\epsilon}^T(t-1)] + \Psi \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1}(\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix} E[\boldsymbol{\eta}(t-3) \boldsymbol{\epsilon}^T(t-1)] \\
&\quad + \Psi \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} E[\bar{\boldsymbol{\eta}}(t-3) \boldsymbol{\epsilon}^T(t-1)] + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} E[\boldsymbol{\epsilon}(t-1) \boldsymbol{\epsilon}^T(t-1)] \\
&= \Psi \mathbf{R}_z \boldsymbol{\epsilon}(t-1) + \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \mathbf{R}_\epsilon \boldsymbol{\epsilon}(t-1). \quad (47)
\end{aligned}$$

The second equality follows from the fact that the zero-mean communication noise vectors are independent of $\boldsymbol{\epsilon}(t-1)$. Scaling (47) by λ yields the desired result.

E. Structure of matrices \mathbf{P}_α , \mathbf{P}_β , $\mathbf{R}_{\bar{\eta}}$, \mathbf{R}_η , $\mathbf{R}_{\bar{\eta}_\lambda}$, and $\mathbf{R}_{\eta_\lambda}$: In order to relate the noise supervectors $\boldsymbol{\eta}_\alpha(t)$ and $\boldsymbol{\eta}_\beta(t)$ with $\boldsymbol{\eta}(t)$ in (26), introduce two $Jp \times (\sum_{j=1}^J |\mathcal{N}_j|)p$ matrices $\mathbf{P}_\alpha := [\mathbf{p}_1 \dots \mathbf{p}_J]^T$ and $\mathbf{P}_\beta := [\mathbf{p}'_1 \dots \mathbf{p}'_J]^T$. The $(\sum_{j=1}^J |\mathcal{N}_j|)p \times p$ submatrices \mathbf{p}_j , \mathbf{p}'_j are given by $\mathbf{p}_j := [(\mathbf{p}_{j,1})^T \dots (\mathbf{p}_{j,J})^T]^T$ and $\mathbf{p}'_j := [(\mathbf{p}'_{j,1})^T \dots (\mathbf{p}'_{j,J})^T]^T$, with $\mathbf{p}_{j,r}$, $\mathbf{p}'_{j,r}$ defined for $r = 1, \dots, J$ as

$$\mathbf{p}_{j,r}^T := \begin{cases} \frac{c}{4} \mathbf{b}_{|\mathcal{N}_r|, r(j)}^T \otimes \mathbf{I}_p & \text{if } j \in \mathcal{N}_r \\ \mathbf{0}_{p \times |\mathcal{N}_r|p} & \text{if } j \notin \mathcal{N}_r \end{cases}, \quad (\mathbf{p}'_{j,r})^T := \begin{cases} \frac{c}{4} \mathbf{1}_{1 \times |\mathcal{N}_r|} \otimes \mathbf{I}_p & \text{if } r = j \\ \mathbf{0}_{p \times |\mathcal{N}_r|p} & \text{if } r \neq j \end{cases}.$$

Note that $r(j) \in \{1, \dots, |\mathcal{N}_r|\}$ denotes the order in which $\boldsymbol{\eta}_j^r(t)$ appears in $\{\boldsymbol{\eta}_{j'}^r(t)\}_{j' \in \mathcal{N}_r}$ [cf. (26)]. It is straightforward to verify that $\boldsymbol{\eta}_\alpha(t) = \mathbf{P}_\alpha \boldsymbol{\eta}(t)$ and $\boldsymbol{\eta}_\beta(t) = \mathbf{P}_\beta \boldsymbol{\eta}(t)$.

Moving on to characterize the structure of $\mathbf{R}_{\bar{\eta}}$ and \mathbf{R}_η , from (21) and recalling that communication noise vectors are assumed uncorrelated in space [cf. (a3)], it follows that

$$\mathbf{R}_{\bar{\eta}} = \text{bdiag} \left(\sum_{j' \in \mathcal{N}_1 \setminus \{1\}} \mathbf{R}_{\eta_{1,j'}}, \dots, \sum_{j' \in \mathcal{N}_J \setminus \{J\}} \mathbf{R}_{\eta_{J,j'}} \right).$$

Likewise, it follows from (26) that \mathbf{R}_η is a block diagonal matrix with a total of $\sum_{j=1}^J |\mathcal{N}_j|$ diagonal blocks of size $p \times p$, namely

$$\mathbf{R}_\eta = \text{bdiag} \left(\{\mathbf{R}_{\eta_{j',1}}\}_{j' \in \mathcal{N}_1}, \dots, \{\mathbf{R}_{\eta_{j',J}}\}_{j' \in \mathcal{N}_J} \right).$$

Note also that the blocks $\mathbf{R}_{\eta_{j,j}} = \mathbf{0}_{p \times p}$ for all $j \in \mathcal{J}$, since a sensor does not communicate with itself. In both cases, the block diagonal structure of the covariance matrices is due to the spatial uncorrelatedness of the noise vectors.

What is left to determine is the structure of $\mathbf{R}_{\bar{\eta}_\lambda}$ and $\mathbf{R}_{\eta_\lambda}$. From (30) one readily obtains

$$\mathbf{R}_{\bar{\eta}_\mu} = \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix} \mathbf{R}_{\bar{\eta}} \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} \\ \mathbf{0}_{Jp \times Jp} \end{bmatrix}^T, \quad \mathbf{R}_{\eta_\mu} = \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} (\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix} \mathbf{R}_\eta \begin{bmatrix} \mathbf{R}_{h,\lambda}^{-1} (\mathbf{P}_\alpha - \mathbf{P}_\beta) \\ \mathbf{C} \end{bmatrix}^T. \quad (48)$$

REFERENCES

- [1] A. Bertrand, M. Moonen, and A. H. Sayed, "Diffusion bias-compensated RLS estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 59, 2011. [Online]. Available: [http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=\\$&\\$arnumber=5975252](http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=$&$arnumber=5975252)
- [2] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, 2nd ed. Athena-Scientific, 1999.
- [3] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, pp. 1865–1877, May 2008.
- [4] F. S. Cattivelli and A. H. Sayed, "Hierarchical diffusion algorithms for distributed estimation," in *Proc. of the Wrkshp. on Statistical Signal Proc.*, Cardiff, Wales, Aug. 2009, pp. 537–540.
- [5] —, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, pp. 1035–1048, Mar. 2010.
- [6] —, "Diffusion strategies for distributed Kalman filtering and smoothing," *IEEE Trans. Autom. Contr.*, vol. 55, pp. 2069–2084, 2010.
- [7] —, "Analysis of spatial and incremental LMS processing for distributed estimation," *IEEE Trans. Signal Process.*, vol. 59, pp. 1465–1480, Apr. 2011.
- [8] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Adaptive robust distributed learning in diffusion sensor networks," *IEEE Trans. Signal Process.*, vol. 59, pp. 4692–4707, Oct. 2011.
- [9] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. on Information Theory*, vol. 46, pp. 388–404, Mar. 2000.
- [10] Y. Hatano, A. K. Das, and M. Mesbahi, "Agreement in presence of noise: pseudogradients on random geometric networks," in *Proc. of the 44th Conf. on Dec. and Contr.*, Seville, Spain, Dec. 2005, pp. 6382–6387.
- [11] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: link failures and channel noise," *IEEE Trans. Signal Process.*, vol. 57, pp. 355–369, Jan. 2009.
- [12] L. Li, C. G. Lopes, J. Chambers, and A. H. Sayed, "Distributed estimation over an adaptive incremental network based on the affine projection algorithm," *IEEE Trans. Signal Process.*, vol. 58, pp. 151–164, Jan. 2010.
- [13] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 55, pp. 4064–4077, Aug. 2007.
- [14] —, "Diffusion least-mean squares over adaptive networks: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 56, pp. 3122–3136, Jul. 2008.
- [15] G. Mateos, I. D. Schizas, and G. B. Giannakis, "Distributed recursive least-squares for consensus-based in-network adaptive estimation," *IEEE Trans. Signal Process.*, vol. 57, pp. 4583–4588, Nov. 2009.
- [16] —, "Performance analysis of the consensus-based distributed LMS algorithm," *EURASIP Journal on Advances in Signal Processing*, Dec. 2009, article ID 981030.
- [17] A. Nedic and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM Journal on Optimization*, vol. 12, pp. 109–138, Jan. 2001.
- [18] C. H. Papadimitriou, *Computational Complexity*. Addison-Wesley, 1993.
- [19] M. G. Rabbat and R. D. Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE Journal on Sel. Areas In Comm.*, vol. 23, pp. 798–808, 2005.
- [20] M. G. Rabbat, R. D. Nowak, and J. A. Bucklew, "Generalized consensus computation in networked systems with erasure links," in *Proc. of the Wrkshp. on Signal Proc. Adv. in Wireless Communications*, New York, NY, Jun. 2005, pp. 1088–1092.

- [21] S. S. Ram, A. Nedic, and V. V. Veeravalli, "Stochastic incremental gradient descent for estimation in sensor networks," in *Proc. of 41st Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, 2007, pp. 582–586.
- [22] A. Ribeiro, I. D. Schizas, S. I. Roumeliotis, and G. B. Giannakis, "Kalman filtering in wireless sensor networks: Incorporating communication cost in state estimation problems," *IEEE Control Syst. Mag.*, vol. 30, pp. 66–86, Apr. 2010.
- [23] A. H. Sayed, *Fundamentals of Adaptive Filtering*. John Wiley & Sons, 2003.
- [24] A. H. Sayed and C. G. Lopes, "Distributed recursive least-squares over adaptive networks," in *Proc. of 40th Asilomar Conf. On Signals, Systems and Computers*, Pacific Grove, CA, Oct./Nov. 2006, pp. 233–237.
- [25] I. D. Schizas, G. Mateos, and G. B. Giannakis, "Consensus-based distributed recursive least-squares estimation using ad hoc wireless sensor networks," in *Proc. of 41st Asilomar Conf. On Signals, Systems and Computers*, Pacific Grove, CA, Nov. 2007, pp. 386–390.
- [26] —, "Distributed LMS for consensus-based in-network adaptive processing," *IEEE Trans. Signal Process.*, vol. 57, pp. 2365–2381, Jun. 2009.
- [27] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc WSNs with noisy links - part I: Distributed estimation of deterministic signals," *IEEE Trans. Signal Process.*, vol. 56, pp. 350–364, Jan. 2008.
- [28] V. Solo, "The stability of LMS," *IEEE Trans. Signal Process.*, vol. 45, pp. 3017–3026, Dec. 1997.
- [29] V. Solo and X. Kong, *Adaptive Signal Processing Algorithms: Stability and Performance*. Prentice Hall, 1995.
- [30] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 58, pp. 4795–4810, Sep. 2011.
- [31] P. Tseng, "Applications of a splitting algorithm to decomposition in convex programming and variational inequalities," *SIAM Journal on Control and Optimization*, vol. 29, pp. 119–138, Jan. 1991.
- [32] S.-Y. Tu and A. H. Sayed, "Mobile adaptive networks," *IEEE Sel. Topics Signal Process.*, vol. 5, pp. 649–664, Aug. 2011.
- [33] J.-J. Xiao, A. Ribeiro, T. Luo, and G. B. Giannakis, "Distributed compression-estimation using wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 23, pp. 27–41, Jul. 2006.
- [34] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, pp. 65–78, Sep. 2004.
- [35] N. R. Yousef and A. H. Sayed, "A unified approach to the steady-state and tracking analysis of adaptive filters," *IEEE Trans. Signal Process.*, vol. 49, pp. 314–324, Feb. 2001.

$$\begin{aligned}
 \mathbf{R}_{y_{1,j}}(t) &= [\mathbf{R}_y(t)]_{11,j} \in \mathbb{R}^{p \times p} \\
 \mathbf{R}_y(t) &= \begin{pmatrix} \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \\ \text{---} \end{pmatrix} \in \mathbb{R}^{2Jp \times 2Jp} \\
 \mathbf{R}_{y_1}(t) &= [\mathbf{R}_y(t)]_{11} \in \mathbb{R}^{Jp \times Jp}
 \end{aligned}$$

Fig. 1. The covariance matrix $\mathbf{R}_y(t)$ and some of its inner submatrices that are relevant to the performance evaluation of the D-RLS algorithm.

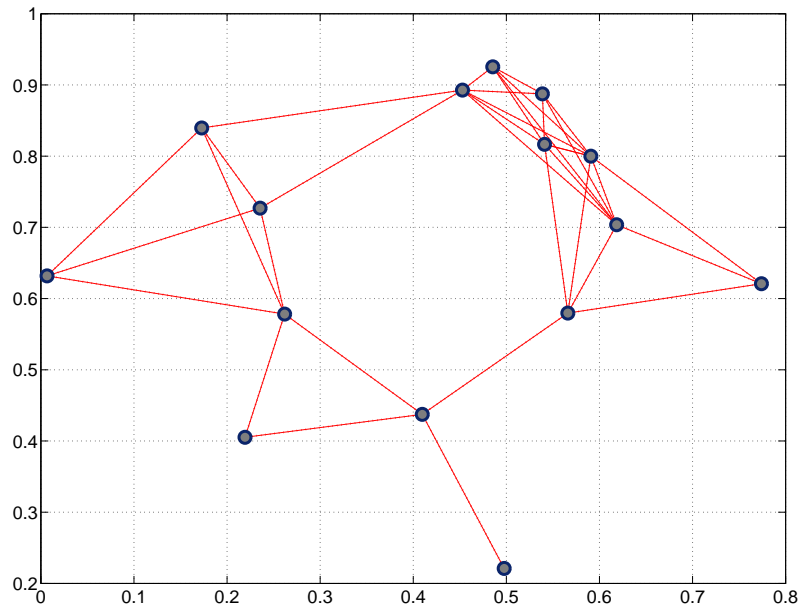


Fig. 2. An ad hoc WSN with $J = 15$ sensors, generated as a realization of the random geometric graph model on the unity square, with communication range $r = 0.3$.

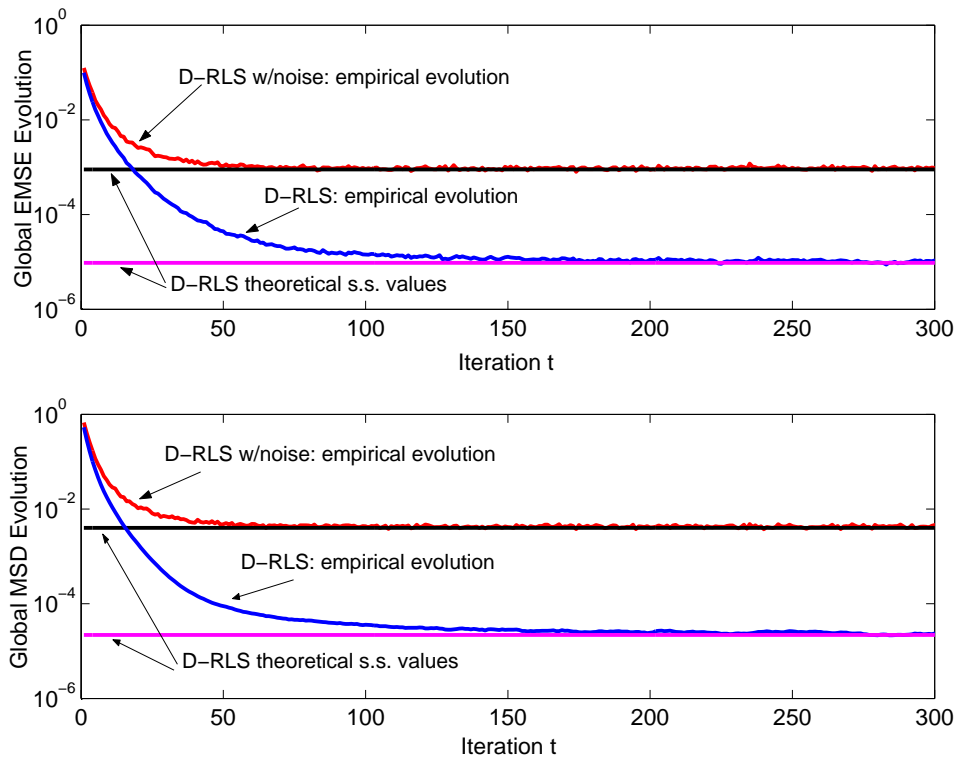


Fig. 3. Global steady-state performance evaluation. D-RLS is ran with ideal links and when communication noise with variance $\sigma_\eta^2 = 10^{-1}$ is present.

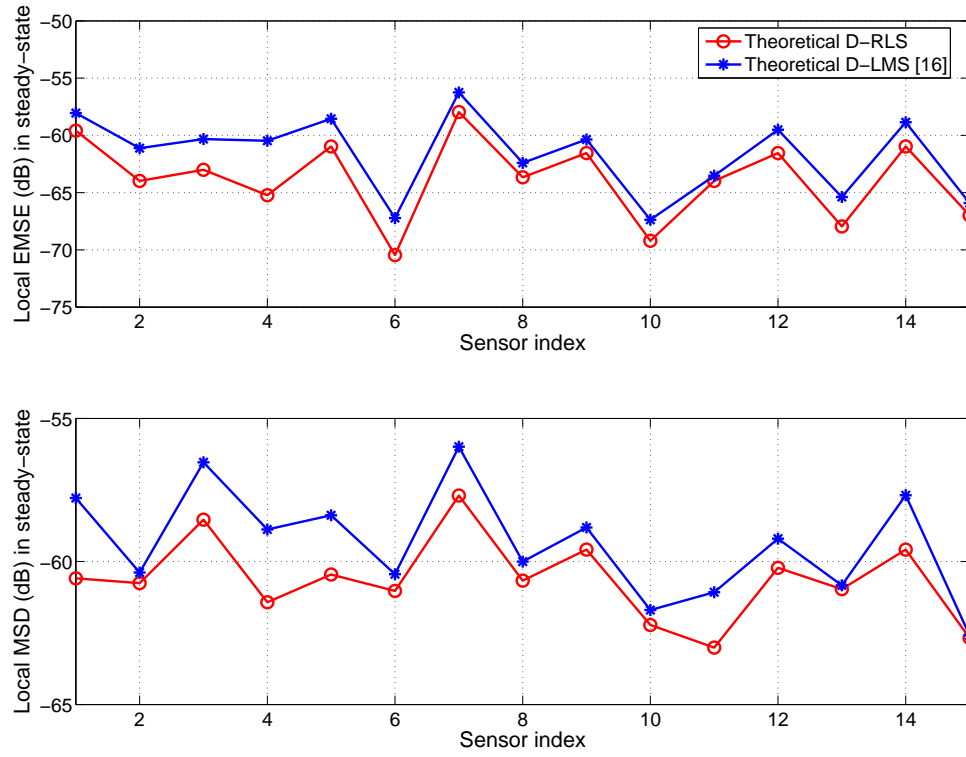


Fig. 4. Local steady-state performance evaluation. D-RLS is compared to the D-LMS algorithm in [16].